YOUR OFFICIAL RASPBERRY PI MAGAZINE

The MagPi

Issue 133 | September 2023 | magpi.cc

The official Raspberry Pi magazine

Emulate Everything

Start Gaming!

- Make an all-in-one console
- Recreate classic computers
- Where to get your games

2023 RASPBERRY PI LTD

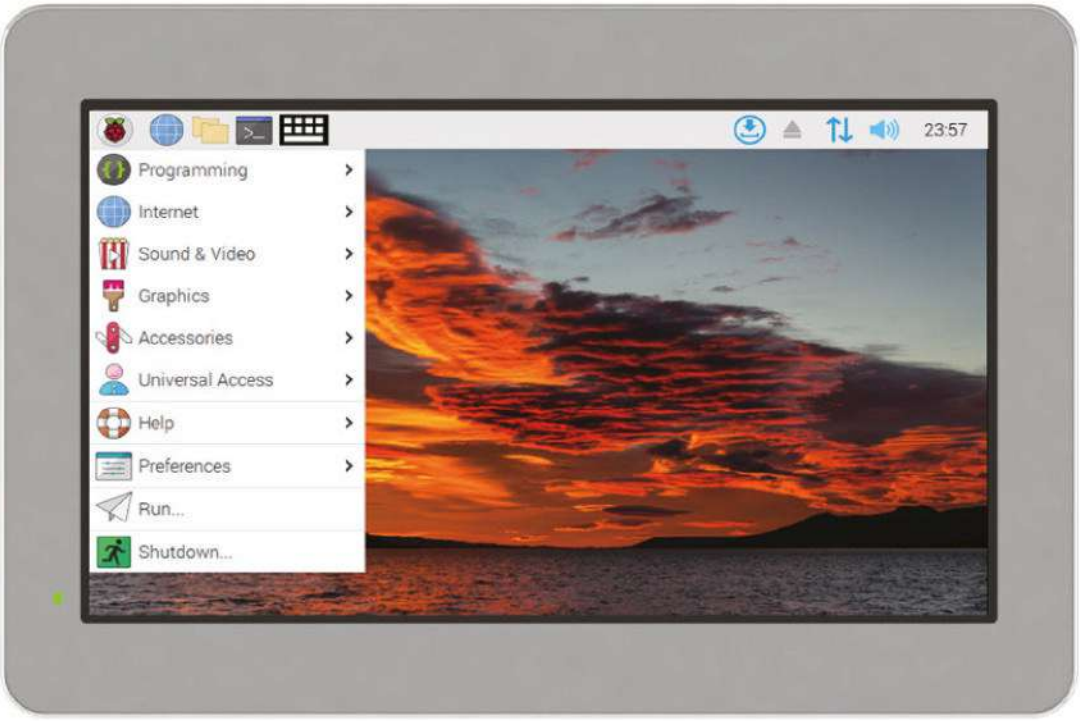
£5.99

9 772051 998001 09

TUTORIALS! REBARTENDER • PONG • CAMERA CONTROL

UL LISTED

Industrial Raspberry Pi



ComfilePi

The ComfilePi is a touch panel PC designed with high-tolerant components and no moving parts for industrial applications. It features a water-resistant front panel, touchscreen, color LCD (available in various sizes), RS-232, RS-485, Ethernet, USB, I2C, SPI, digital IO, battery-backed RTC (real-time clock), and piezo buzzer.

Use the rear-panel 40-pin GPIO header to expand its features and capabilities with additional I/O boards. The ComfilePi is UL Listed and employs Raspberry Pi Compute Module.

POWERED BY Raspberry Pi

Visit www.comfiletech.com

© copyright COMFILE Technology, Inc. ALL RIGHTS RESERVED

COMFILE TECHNOLOGY

WELCOME
The MagPi 133

Retro computing enables us to experience a time when computers were more straightforward and more challenging. The modern technology stack is so detailed that it's hard to fully comprehend, whereas older computers are much easier to figure out.

And the games were terrific. Nostalgia is powerful stuff, and the technical achievement of 1980s games, combined with the level of challenge, is incredible.

Classic computing is a huge hobby. And our resident retro expert, KG Orphanides, has written an Emulate Everything feature (**page 36**). It's packed with information on all the different retro options, the legal status of the various systems, and where to get your retro games.

Build a great system and happy gaming.

Lucy Hattersley Editor

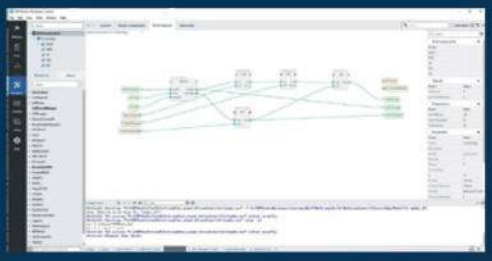
EDITOR

Lucy Hattersley

Lucy is editor of *The MagPi* and is coaxing a 3D printed camera into shape this month.

magpi.cc

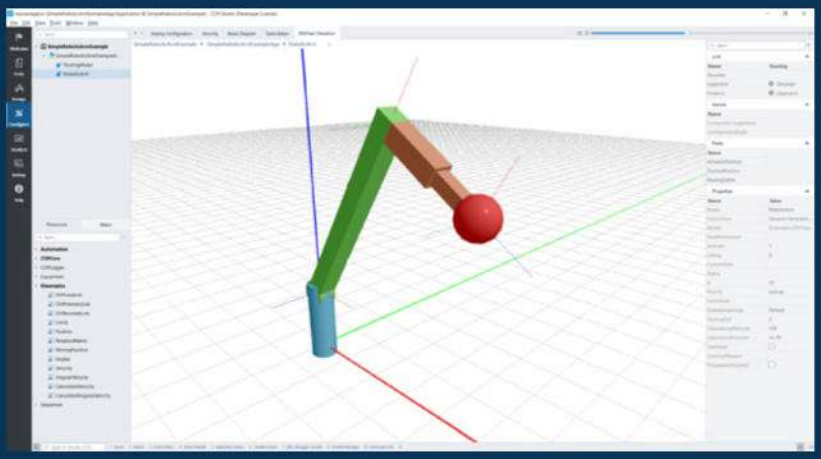
GET A RASPBERRY PI PICO W WITH A SUBSCRIPTION! PAGE 30



No-code

Full-code

HMI for panel PC and Web



Robotic arm with DHChain configurator and visualizer

No-Code

Kinematics and Visualizer

PROFESSIONAL CONTROL SYSTEM DEVELOPMENT TOOL

Home projects made easy.

CDP Studio, a great software development tool for your home projects. Build systems for Raspberry Pi, use C++ or NoCode programming, open source libraries, out of the box support for GPIO, I2C, MQTT, OPC UA and more. Create beautiful user interfaces. Built for industrial control system development, **FREE for home projects**.

cdpstudio.com
Tel: +47 990 80 900 • info@cdptech.com
CDP Technologies AS // Hundsværgata 8, 6008 Ålesund, Norway

CDPStudio

Contents

Issue 133 > September 2023

Cover Feature

36 Emulate Everything

Regulars

32 Case Study: Glacier Research
92 Your Letters
97 Next Month
98 The Final Word

Project Showcases

08 Roktrack
12 Pico Held
14 EPIC Satellite
18 OnnxStream
22 Chip Companion Robot
26 Badgercam

Roktrack

Chip

The MagPi is published monthly by Raspberry Pi Ltd, Maurice Wilkes Building, St. John's Innovation Park, Cowley Road, Cambridge, CB4 0DS, United Kingdom. Publishers Service Associates, 2406 Reach Road, Williamsport, PA, 17701, is the mailing agent for copies distributed in the US and Canada. Application to mail at Periodicals prices is pending at Williamsport, PA. POSTMASTER: Send address changes to The MagPi, c/o Publishers Service Associates, 2406 Reach Road, Williamsport, PA, 17701.

Tutorials

44 Practical Python Part 1
48 Build a Pong Game
54 Control Camera Module settings
60 Make an automated bartender

The Big Feature

64 Work and Learn with Raspberry Pi

Reviews

72 CrowVision
74 Pironman case
76 Spring time is here
82 10 amazing camera projects
84 Learn Linux

Community

86 HannahMakes interview
88 This Month in Raspberry Pi
94 Community Events

Control Camera Module settings

Make an automated bartender

Work and Learn with Raspberry Pi

10 amazing camera projects

HannahMakes interview

WIN 1 OF 5 RASPBERRY PI 4 **8GB**

DISCLAIMER: Some of the tools and techniques shown in The MagPi magazine are dangerous unless used with skill, experience, and appropriate personal protection equipment. While we attempt to guide the reader, ultimately you are responsible for your own safety and understanding the limits of yourself and your equipment. Children should be supervised. Raspberry Pi Ltd does not accept responsibility for any injuries, damage to equipment, or costs incurred from projects, tutorials or suggestions in The MagPi magazine. Laws and regulations covering many of the topics in The MagPi magazine are different between countries, and are always subject to change. You are responsible for understanding the requirements in your jurisdiction and ensuring that you comply with them. Some manufacturers place limits on the use of their hardware which some projects or suggestions in The MagPi magazine may go beyond. It is your responsibility to understand the manufacturer's limits.

TRUST
STARTS HERE

From genuine, manufacturer-warrantied components to millions of in-stock parts shipped same day, be confident Digi-Key will get you what you need—when you need it.

Visit **digikey.co.uk** today, or call 0800 587 0991.

Digi-Key
ELECTRONICS

Digi-Key is a franchised distributor for all supplier partners. New products added daily. Digi-Key and Digi-Key Electronics are registered trademarks of Digi-Key Electronics in the U.S. and other countries. © 2023 Digi-Key Electronics, 701 Brooks Ave. South, Thief River Falls, MN 56701, USA

ECIA MEMBER

Roktrack – Pylon-guided mower

Here's a robot that takes grass and weed cutting to the next level... literally. **Nicola King** wonders at this weed assassin

MAKER

Yuta Suito

A software engineer, Yuta previously worked in Tokyo, but now lives in a very mountainous area. He wanted to create an outdoor tool to improve people's lives.

hackaday.io/yutasuito

Warning!
Mower safety

Fast-spinning lawnmower blades are very dangerous and can also eject debris at high speed.

magpi.cc/mowersafety

Living in a mountainous region of Japan has proved inspiring for maker Yuta Suito. The local area has many rice terraces but, as he explains, "With the ageing of the farmers, this beautiful landscape will disappear in about ten years. Many farmers say that the hardest part of maintaining the terraced fields is weeding." Determined to try and help, Yuta wanted to "solve this problem by providing mobility (with no need to bury boundary wires underground) and intuitive operation (in reality, many elderly people cannot use smartphones or computers)."

The result is the lightweight, solar-powered Roktrack pylon-guided mower, and it's perfect for rough terrain. Prompted by a road construction site, Yuta noticed that "by simply enclosing the area with traffic cones, the people inside would understand the scope of the work. I was inspired by the idea that computers could do the same thing." Applying this logic to his project, Yuta's mower is able to mow a range of coned-off areas, with a 200 m² area, for example, taking around two hours. Not only that, but it's equipped with a speaker, so "if you are in the immediate vicinity, an audible voice will tell you when the mowing is done!"

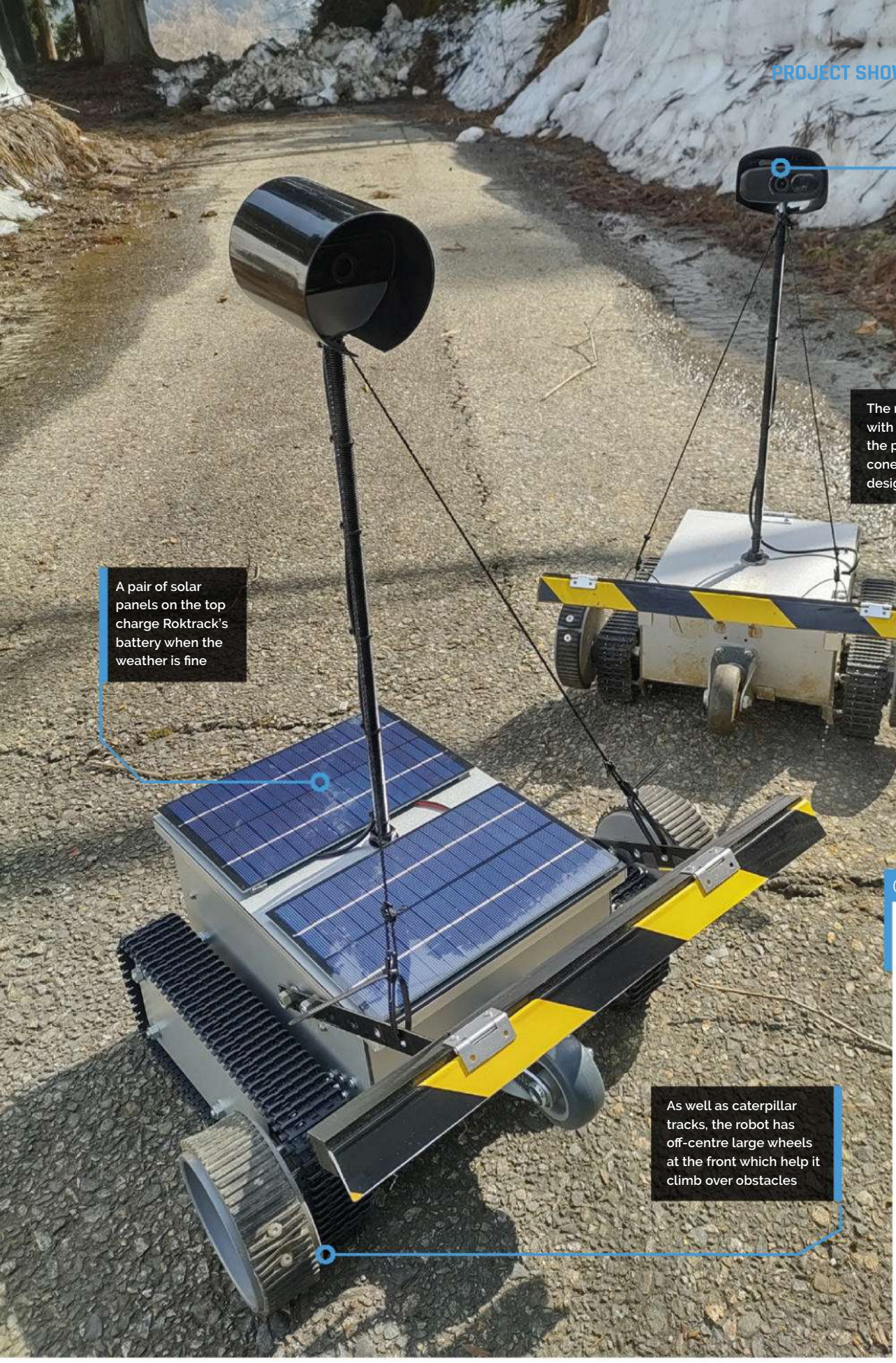
Traffic cones act as a visual guide for Roktrack, helping it to work its way around an identified area

The word in weeding

With software written by Yuta, Roktrack is programmed to head towards a pylon (or cone) which it recognises through a camera image, and it then starts working its way towards the next one. Equipped with two blades, it cuts backs grass and weeds as it goes, leaving them on the ground. It's rain-proof and has eccentric front wheels which allow it to traverse less hospitable terrain than you'd find in your average garden. In addition, object detection means that two mowers operating in the same area won't collide.

Yuta used a Raspberry Pi 3 Model A+ to control Roktrack for temperature and power consumption reasons. "I wanted to cool the Raspberry Pi only with the fan provided on [Roktrack], not with a dedicated fan for Raspberry Pi. In addition, since this unit uses only one LiFePO4 (4S, 12.6 V) power supply for the control and drive systems, it was necessary to step down the power supply for Raspberry Pi."

To start mowing, the user just presses the power button – or they can use an Android app, where communication between the robot and the app relies on Bluetooth Low Energy (BLE). Impressively, the app can instruct the mower to make an emergency stop, switch mode instructions, and, as Yuta details, "the coordinates



The robot is equipped with a camera to spot the pylons (traffic cones) marking the designated area

A pair of solar panels on the top charge Roktrack's battery when the weather is fine

As well as caterpillar tracks, the robot has off-centre large wheels at the front which help it climb over obstacles

Quick **FACTS**

- Yuta has spent around two years on this project...
- ...with more than one year spent building software using image recognition
- The raw materials for the current prototype cost around £275...
- ...with trial and error leading to a total cost of around £1650
- Yuta's also considering creating a winter snow removal system

An underside view of the robot shows its two rotary blades which enable pebbles to be ejected

of the past mowing point are recorded by the Android [device]'s GPS, and the current vegetation (weed growth) can be remotely sensed using Sentinel satellite images."

Finding solutions

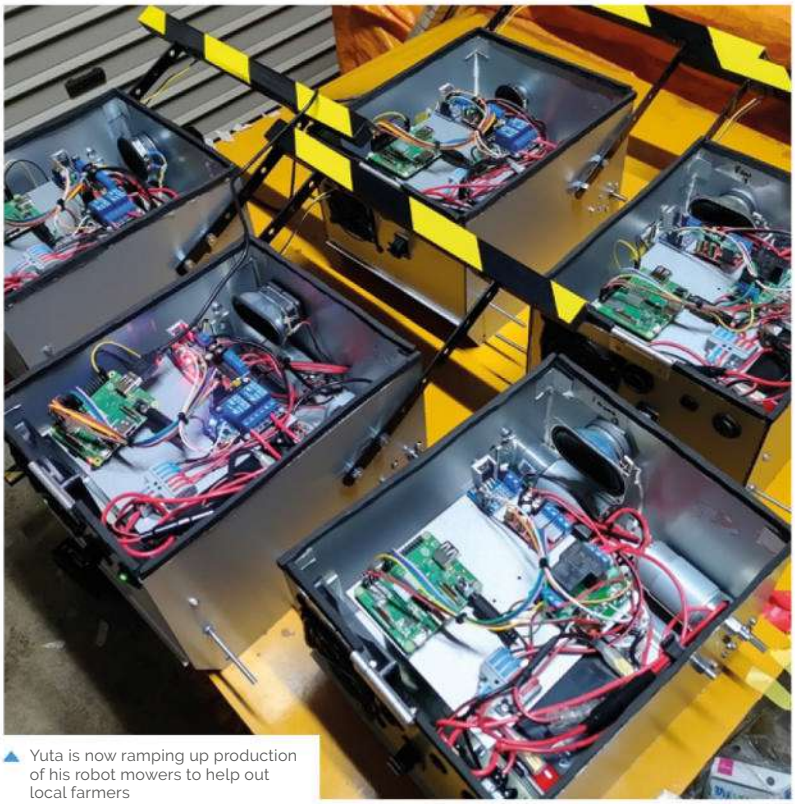
Testing Roktrack was challenging, says Yuta. "It was very difficult to test the system until I could get it to work stably." He once took his eyes off of it while testing, and the robot tumbled down a cliff, but his conscientious hard work, testing thoroughly, and solving problems "one by one" now means, "I just leave it on and have a cup of tea in the house, and the weeds are cut." Yuta is currently developing what he calls "parallel operation with multiple units," and, by the end of the year, he hopes to experiment with up to six mowers working on several hundred square metres at a time.

The building of a robotic mower like this is probably not a suitable beginner project: for one thing the creation of the chassis gave Yuta some issues initially, but then he discovered a sheet metal service (**PCBWay.com**), which was able to send him "precisely machined" aluminium sheet for his project, and this solved a number of problems. "I don't think it would be easy to make it from scratch, as it would require a lot of tools and the number of parts is very large," he notes. "However, you can find the 2DCAD [file] of the body on Hackaday (**magpi.cc/roktrack**). If you use an aluminium sheet metal service, you can reduce the difficulty of the work."

A life changer

Pleasingly, Yuta's project has brought people together. "The field robots are very eye-catching and provide an opportunity to talk to many people, including passers-by," he notes. More than that,

A couple of Roktrack robots in action on tricky terrain



Yuta is now ramping up production of his robot mowers to help out local farmers

> "I just leave it on, and have a cup of tea in the house, and the weeds are cut"

the project appears to have struck a chord with people and, when presented to the Japanese maker community, it became a "hot topic." Interviews followed and his local government has agreed to support the project, and five additional units will now be built to mow vacant lots in the community. "Together with the two I have built so far, I will have a total of seven units. I feel that this project is definitely starting to change my life," declares Yuta.

Japan is not the only part of the world with rice terraces and the inherent issues that rice farmers are facing, as other parts of Asia, where rice-eating is an important part of the culture, face potentially the same problems. Happily, Yuta has received well-deserved interest from potential investors, including "venture companies, mower manufacturers, and agricultural corporations. It would be great if this project becomes a business." Ingenuity and resourcefulness at its best – we wish him well!

Grass me up

01 Yuta's casing needed to be strong to weather the terrain, so aluminium sheet metal was chosen. He then installed motors, fans, and switches.

02 Yuta converted Raspberry Pi OS on his 3A+ to read-only. He also allocated it more memory so that it could handle the task (**magpi.cc/roktrackreadonly**). The mowing program is automatically started by systemd, and starts mowing on its own.

03 Yuta attached solar panels to the top of the fully assembled robot with double-sided tape. The battery can be charged from the panels. Under solar power, Roktrack can operate for up to an hour a day, if the weather is good.

Pico Held

David Crookes takes a look at a device based around Raspberry Pi Pico that emulates the Sega Mega Drive and lets you play homebrew games

MAKER

Daniel Kammer

Daniel is a physicist working as a teacher in Germany. He loves embedded and retro systems because he enjoys the challenge posed by limited resources, as well as their huge DIY potential.

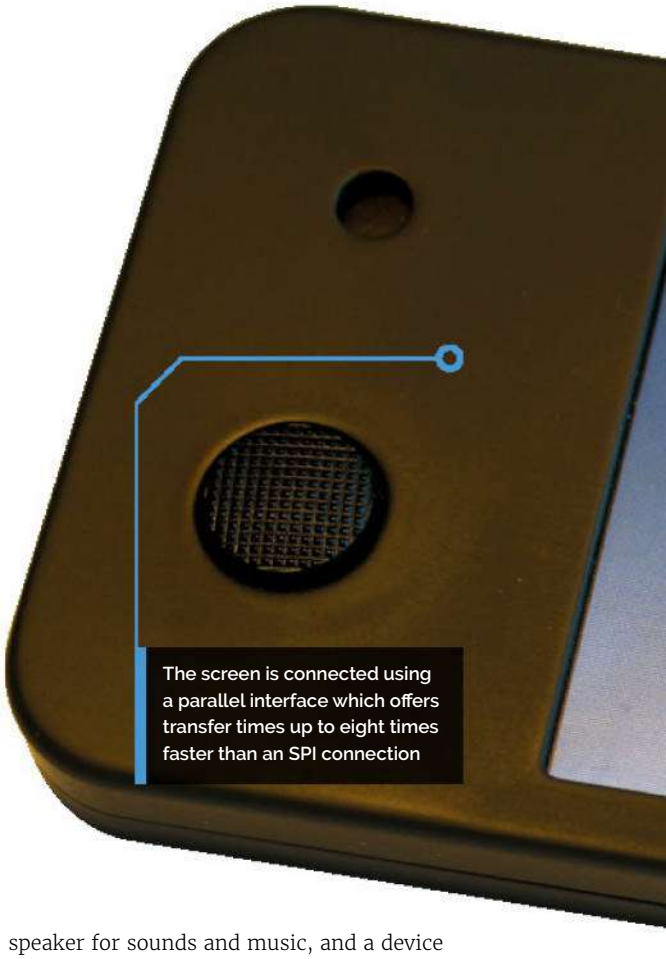
magpi.cc/pplib

Although there are many handheld retro game consoles based around Raspberry Pi devices, we still want to play with more. They help to keep the memories of old machines and titles alive, while providing the perfect opportunity to dive into a project and learn more about what is going on under-the-hood.

It's this desire to understand that has driven Daniel Kammer to create Pico Held, a beautiful, open-source handheld with a Raspberry Pi Pico board at its heart. "There is so much you can learn from creating DIY projects," he says. "By making it open-source, it also gives others the opportunities to learn – maybe they'll help to improve Pico Held in the future."

Hold fire

Before getting down to work, Daniel came up with some goals. "I wanted an appealing design, a screen big enough to play games on, a nice speaker for sounds and music, and a device powered by Raspberry Pi Pico – an amazing piece of hardware with a balanced set of features," he explains.



The screen is connected using a parallel interface which offers transfer times up to eight times faster than an SPI connection

There were clear reasons for this. "A small screen and subpar speaker kind of defeats the whole point of having a gaming device in the first place," he notes. This prompted Daniel to hunt for the largest, low-cost screen possible. "A 3.2-inch display was the biggest IL19341 I could get on AliExpress," he says, but the goals posed a host of challenges.

Since Daniel custom-made a PCB and 3D-printed a case, he needed everything to fit. "I had to align the button holes to the PCB contacts, make sure the LCD was nicely centred, and the analogue stick was fitting," he explains. "I also found it quite challenging not to exceed a thickness of 12mm. For every flaw, I made a new PCB and new case!"

In the end, sacrifices needed to be made. "My prototype had four buttons, but I decided to go for three and use an analogue stick instead of a D-pad," he says. "I also wanted to design the Pico Held in a way that others could build it too. That was a big constraint."

Daniel says homebrew game developers would be able to find a way to cope with three buttons. "DIY is about creativity, right?", he says

To be this good

Developing the software proved even more challenging. Daniel wanted his handheld to play retro games, so he modified a Sega Mega Drive

A hello to
The MagPi
As well as a 3W speaker, there's an analogue control stick, three buttons, and an on/off switch
The 3.2-inch screen has a resolution of 320 × 240 pixels and it sits in a case that's 136 × 60 × 12 mm

emulator called Gwenesis by bzhxx. "I had to fiddle with the endianness [the order in which bytes are stored] and the RAM/ROM handling," he recalls. "It was pretty frustrating because virtually all errors were hard faults."

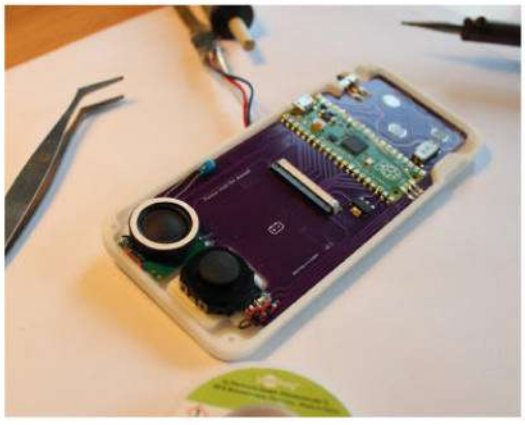
The result has been an emulator running Mega Drive games more slowly than intended, but Daniel says lots of games still remain playable.

> A small screen defeats the whole point of having a gaming device in the first place

"Sound was a bit trickier: I had to make the sound generation per frame/per line rather than per cycle, and that causes the sound to be flawed in a lot of games." Daniel has also been able to run a NES emulator on his device: "It runs at full speed," he tells us.

More than that, Daniel wants people to create their own games. "The Pico Held's software library was written with the idea of allowing people to create classic pixel art games with 256 colours," he reveals. "Creating the blitter and the tile map blitter in the software library for this purpose kept me busy for days."

Yet, it's been worth the effort. "Some have said the analogue stick isn't well-suited for this kind of game device, and some want four buttons," he says, confessing that he isn't actually really into gaming. "But the reactions have been surprisingly positive. I was really glad about that!"



Raspberry Pi Pico works well as a console emulator. RP2040 Plus can be used instead to provide 4 or 16 MB of flash memory

Quick FACTS

- It runs a Mega Drive emulator
- A simple library does the hardware handling
- The library is Arduino-based and beginner-friendly
- You can create your own pixel art games
- The project is entirely open-source

EPIC Satellite

NASA's EPIC Satellite takes beautiful photographs, which one Raspberry Pi user was keen to display at home. **Rosie Hattersley** was intrigued

MAKER

Matt Gray

Matt is a radio and TV broadcast engineer, as well as a keen YouTuber. He loves the problem-solving and creative side of electronics and engineering.

mattg.co.uk

NASA's Deep Space Climate Observatory (DSCOVR) is a satellite that sits between the Earth and the sun, where it is ideally placed for solar research, and provides early warnings of solar winds (that can affect power grids and communication systems).

DSCOVR also hosts a camera called EPIC (Earth Polychromatic Imaging Camera) which is positioned so it can always see the sunlit side of Earth. "EPIC takes beautiful photos of the planet every day, and I wanted a way of displaying the latest photos in my home," explains renowned YouTuber MattGrayYES. Since NASA allows you to access their EPIC photos from your code using their Blue Marble API (**magpi.cc/epicapi**), it was clearly feasible.

When Matt saw the 2.1-inch HyperPixel Round Touch Display in the Pimoroni shop (**magpi.cc/hyperpixelround**), he decided it looked perfect. "This screen is designed for use with Raspberry Pi, so I used a Raspberry Pi Zero W." Matt explains that he needed the wireless capabilities to download the photos from NASA. His approach would be similar to that used in his 'Game Boy Camera Fast Wifi Adapter' project (featured in *The MagPi* issue #110, **magpi.cc/110**), which saw Matt find a way to pull the photos from the handheld console onto his smartphone.

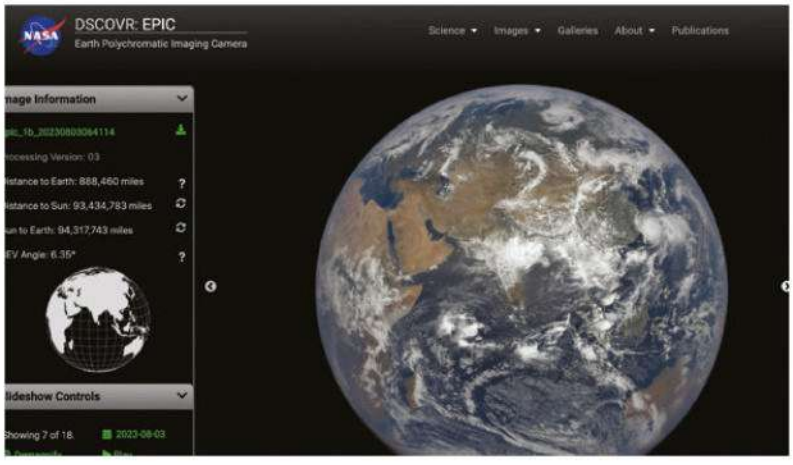
Rounded approach

Although Raspberry Pi Zero W is "way overpowered for the job," Matt chose it because it offers a quick and easy way for him to bring the project into being. "As always, with anything I've made with a Raspberry Pi, the hard bit for me was getting the code to run by itself at startup without any warnings or windows popping up over the top." Matt also praises Pimoroni, who "have done the hard work of making the round screen so easy to use." An alternative option would have been to use a microcontroller and a different round screen, but this would have taken far longer, as so many aspects would have needed to have been written or designed from scratch.

A Raspberry Pi Zero W provides wireless connectivity and handles the Python code and Blue Marble API data, serving up the incredible images

EPIC Satellite displays the images shared daily by NASA's DSCVR satellite which is located at the Earth-Sun Lagrange

The 3D-printed case holds a Pimoroni HyperPixel 2.1-inch circular screen. Ideal for Earth-shaped imagery



EPIC sends back photos of Earth from its vantage point a million miles away

Quick **FACTS**

- Matt first encountered Arduino and Raspberry Pi at university
- He's been making techie videos for more than 15 years
- Such as a Raspberry Pi Freddie Mercury Thermometer
- He also helps organise the Electromagnetic Field Festival
- This gorgeous project cost roughly £120

▲ Matt's infectiously enthusiastic videos convey his love of space and tech

> The API is named after the famous Blue Marble photo of Earth captured by the Apollo 17 space mission

Another piece of good fortune was that someone else had already created a really good-looking case for the HyperPixel display that would look great sitting on a desk.

Matt wrote a Python script that checks the Blue Marble API for new EPIC photos, downloads them, and then loops showing each one in sequence on

▶ Matt's previous Pi Zero W projects include this Game Boy photo printer

▲ The unusual slide show is a fascinating look at Earth from our galaxy, but definitely far, far away

the screen using the Pygame library. The EPIC Daily Blue Marble API is provided by NASA and provides information about the images that the DSCVOR EPIC equipment collects here: **magpi.cc/epicapi**. Its Earth–Sun Lagrange positioning ensures that it is not just the acronym that is EPIC! It captures unique perspectives of astronomical events, such as lunar transits, using a 2048 × 2048 pixel CCD, coupled to a 30 cm aperture Cassegrain telescope.

Case work

The HyperPixel display was useful on its own, but needed something to hold it at a convenient viewing angle. Matt was about to design his own case, but did a quick check online to see whether anyone else in the community had designed a 3D-printable case for the round screen and Raspberry Pi combo. Happily, he found Cults3D user named 'Printminion' who had designed the perfect case (**magpi.cc/printminion**).

Since Printminion's case was specifically designed for the HyperPixel display, Matt was able to focus on how to make use of NASA's Blue Marble API to get the incredible NASA photos on the screen. The API is named after the famous Blue Marble photo of Earth captured by the Apollo 17 space mission in 1972.

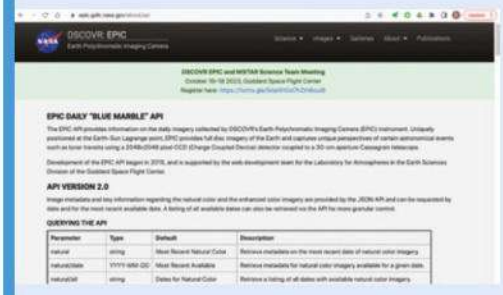
As well as daily images used to form Matt's EPIC slide show, the API can serve up noteworthy images depicting our exploration of near space, and important images such as partial lunar eclipses and transits: **magpi.cc/epicpng**.

Matt has created a making-of video for the EPIC Satellite project on his YouTube channel (**magpi.cc/epicframeyt**). Look out for updates as he begins to make it even more epic, by adding options to view images of the moon and other planets.

Eye spy

01 If you have access to a 3D printer, you can download a template from this link: **magpi.cc/printminion**. Once printed, insert the HyperPixel display.

02 Next, attach Raspberry Pi Zero W to the screen, sandwiching the case between the two. Insert an SD card preloaded with Raspberry Pi OS and preconfigured to automatically connect to a wireless network.



03 Power on and follow the coding instructions Matt provides on his GitHub page for pulling in photos from the Blue Marble API: **magpi.cc/epicgit**. Enjoy the photo slide show!

OnnxStream Stable Diffusion Raspberry Pi Zero 2 W

Creating AI art using a Raspberry Pi Zero 2 W shouldn't work, but a determined software optimiser found a way how, as he tells **Rosie Hattersley**

MAKER

Vito Plantamura

Vito is an enterprise software developer who fell in love with computing aged eight.

vitoplantamura.com

Artificial intelligence has garnered thousands of headlines over the past few years, intriguing and appalling people in roughly equal measure.

AI art generation, whether musical, visual or written, has alarmed many a creator concerned about losing work, or having theirs ripped off.

What we've seen little of, to date, is canny AI generators that can run on a computer with modest processing capabilities. Training AI models requires powerful servers and multiple dedicated GPU (graphics processing unit) boards to create the model that can generate images.

After this training process comes "inference", where a trained model is used to infer a result on a regular computer. Even this takes time on a powerful computer with multiple GBs of RAM.

OnnxStream, however, makes inference possible on a tiny Raspberry Pi Zero with just 512MB RAM and finds a balance between processing power and generation time to turn out impressive imagery via Stable Diffusion.

"Generally major machine learning frameworks and libraries are focused on minimizing inference latency and/or maximizing throughput, all of which are at the cost of RAM usage. So I decided to write a super-small and hackable inference library specifically focused on minimizing memory consumption", says maker Vito Plantamura of how OnnxStream came about.

> "I decided to write a super-small and hackable inference library"

Enterprising guy

Although this is his first big Raspberry Pi project, Vito controls his home TV setup with a Raspberry Pi 4 and sometimes uses one for enterprise-grade tasks "running software jobs that took weeks to execute and that weren't convenient to run on

Images generated by OnnxStream are 512 × 512 pixels. We've upscaled the original images for print

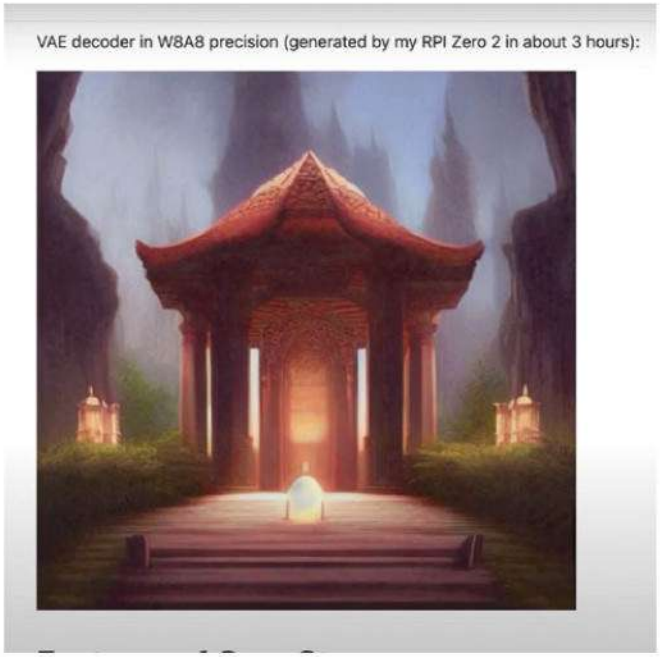
Stable Diffusion uses a billion parameters to create photo-realistic images – and is notoriously graphics-intensive

Maker Vito's OnnxStream code optimises Stable Diffusion's machine learning abilities with results barely distinguishable from a full desktop computer

Quick **FACTS**

- Stable Diffusion uses the LAION dataset
- Which trains its AI engine on a billion parameters
- It usually recommends a computer with a 6GB GPU
- Making Vito's OnnxStream optimisation even more incredible
- He says being able to run a full Linux distro was crucial

More beautifully detailed and rendered OnnxStream AI images



In just 1.5 hours, OnnxStream produced a near identical output on Raspberry Pi Zero 2 to a PC

a desktop computer." He bought a Raspberry Pi Zero 2 W as soon as it was released "because it was too cute!" and says the idea of running Stable Diffusion on such a small device was so intriguing he couldn't resist.

With a day job writing enterprise software, the process of simplifying things to run on Raspberry Pi really appealed, and "allows me to interact on a lower level with computers than what my job allows me to do at the moment."

Discover what AI makes of 1980s pop culture by requesting it generate a popstar of the time

Power play

While his OnnxStream project is intended to run on Raspberry Pi Zero 2, most of the development took place on Vito's Raspberry Pi 4. This meant he had plenty of processing power available to get the code running smoothly, and could focus on optimising it and reducing its requirements knowing that it was fully operational. The process required a number of iterations in order to optimisations the OnnxStream code and, only when the memory consumption dropped below 300MB, did he begin the final tests on Raspberry Pi Zero 2 W.

> "The only limitation is, of course, speed and memory"

The first version of Stable Diffusion for OnnxStream consumed 1.1GB of RAM, far too much for a Raspberry Pi Zero 2 W, but he eventually "managed to get the three models of Stable Diffusion into the memory of the little Zero."

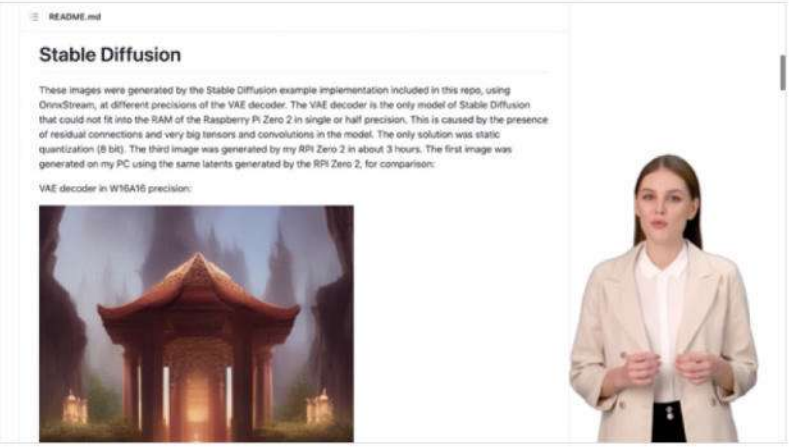
He developed OnnxStream over a five-month period, developing and testing on both Raspberry Pi 4 and Raspberry Pi Zero 2 W. He notes that

Stable Diffusion Playground
Just enter your prompt and click the generate button.
No code required to generate your image!
A delicious ceviche cheesecake slice
Generate Image

▲ You're able to test out Stable Diffusion online through a web interface too - no downloads required

"both run a full Linux operating system [so] compared to a full desktop or server system, there's no limit to what you can do. The only limitation is,of course, speed and memory. That's why running Stable Diffusion on the smaller of the two is such an intriguing idea: being able to overcome one of these two barriers brings Raspberry Pi Zero 2 W closer to a modern desktop system costing two orders of magnitude more."

Vito's detailed GitHub explains that the MAX_SPEED option allows to increase performance by about 10% in Windows, but by more than 50% on the Raspberry Pi. Using this feature, he was able to reduce the generation time on his Raspberry Pi 2 Zero W from three hours down to 1.5. However, he warns that this option consumes much more memory at build time and there is a chance of the executable not working. Should this be the case, he recommends retrying with MAX_SPEED set to OF. Given the complexity of the task OnnxStream.


README.md
Stable Diffusion
These images were generated by the Stable Diffusion example implementation included in this repo, using OnnxStream, at different precisions of the VAE decoder. The VAE decoder is the only model of Stable Diffusion that could not fit into the RAM of the Raspberry Pi Zero 2 in single or half precision. This is caused by the presence of residual connections and very big tensors and convolutions in the model. The only solution was static quantization (8 bit). The third image was generated by my RPI Zero 2 in about 3 hours. The first image was generated on my PC using the same latents generated by the RPI Zero 2, for comparison:
VAE decoder in W16A16 precision:

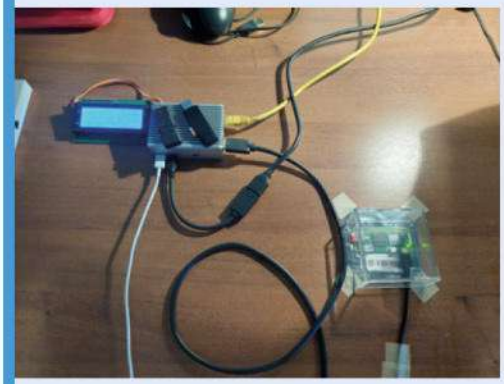
▼ Even the YouTube piece discussing Vito's OnnxStream project uses an AI presenter

Optimise AI imaging

Stable Diffusion Playground
Just enter your prompt and click the generate button.
No code required to generate your image!
Generate Image

01 OnnxStream makes use of XNNPACK (**magpi.cc/xnnpack**) in order to optimise how Stable Diffusion generates AI imagery.

02 Testing was done using Raspberry Pi OS Lite 64-bit, with OnnxStream eventually able to run all but one neural network encoder on Zero 2 W.



03 Vito's GitHub details setting up OnnxStream on a Raspberry Pi with "every KB of RAM needed to run Stable Diffusion: **magpi.cc/onyxstreamgit**.

Chip Bipedal Companion Robot

Chip is accomplished robot maker Kev's take on a cute companion. **Rosie Hattersley** hears his origin story

MAKER

Kevin McAleer

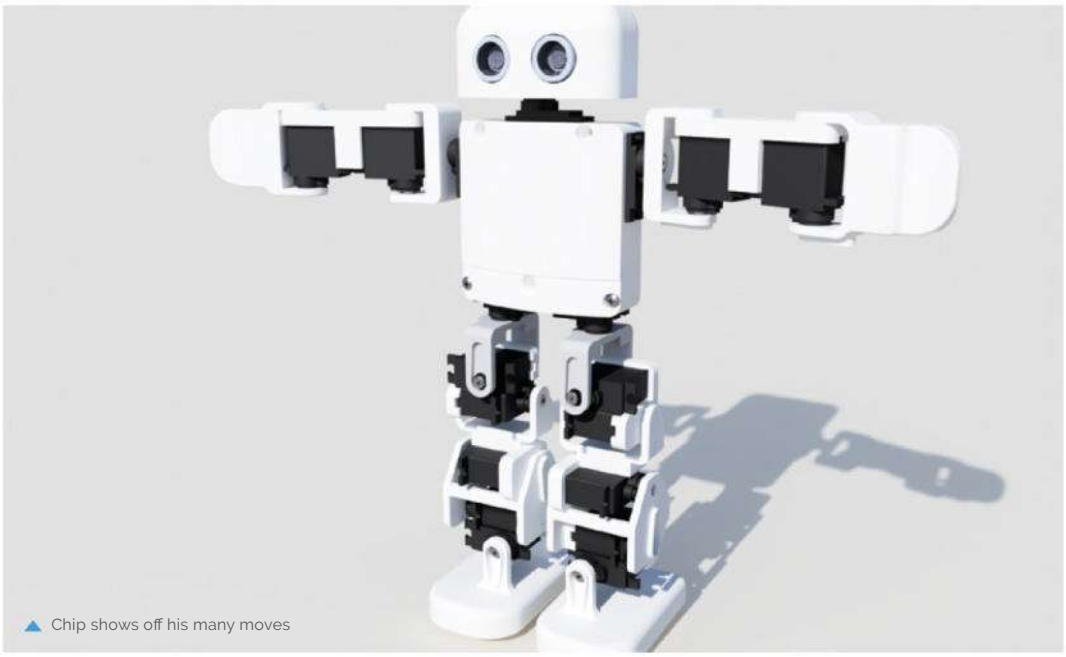
Kevin McAleer is a master of robotics, wowing us nearly weekly with a new robot.

kevsrobots.com

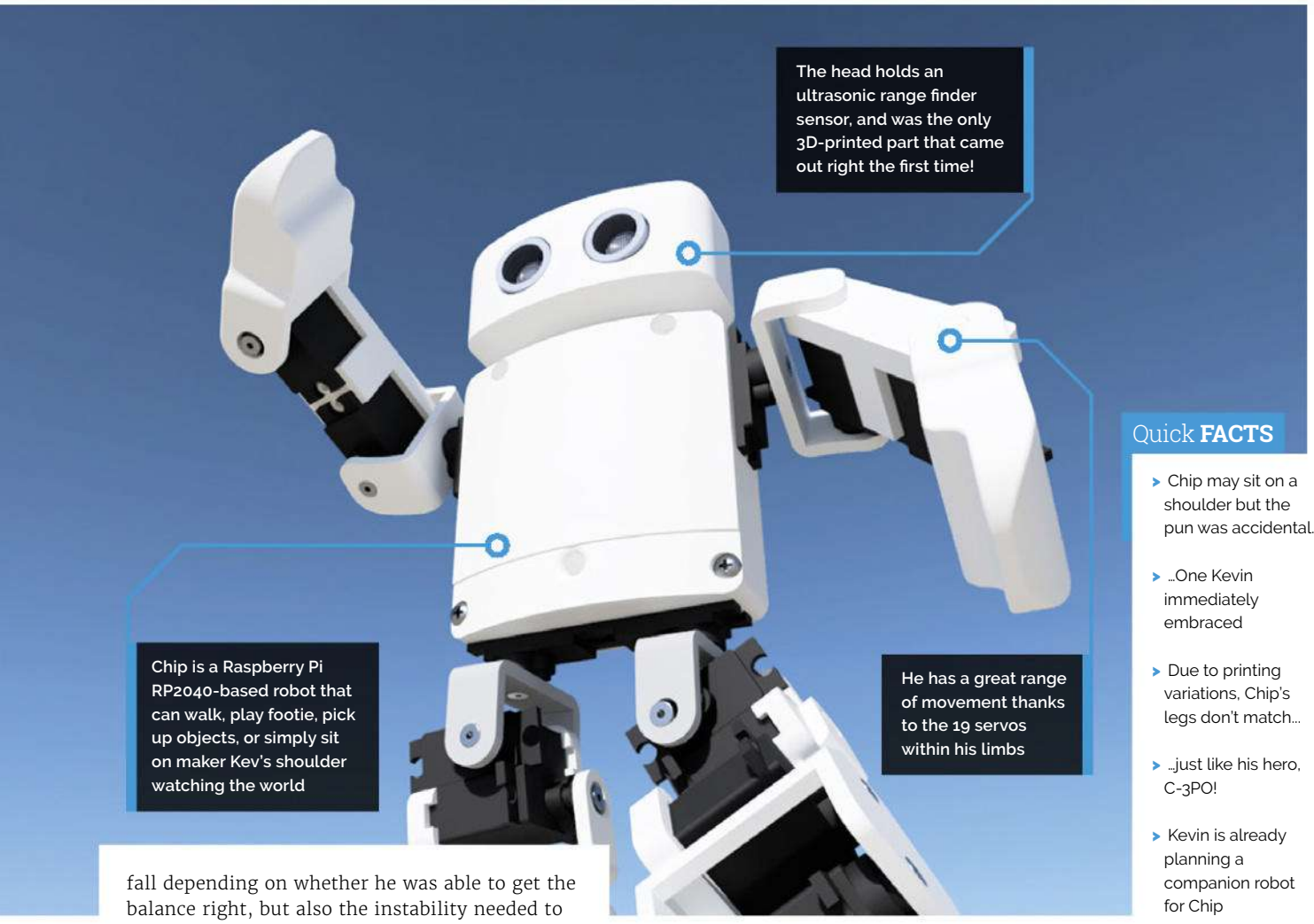
Many readers will be familiar with maker Kevin McAleer's endearing robots and other inventions, either through reading *The MagPi* or enjoying his wonderful YouTube broadcasts which enthuse about all things tech. Earlier this year, he designed and built a Pico-based companion named Bubo 2T. This second companion, Chip (**magpi.cc/chip**), is so-named because it will sit on its maker's shoulder (insert pun here). Whereas Bubo was a social media maven, tweeting out missives, Chip is more of a 'cute interactive humanoid pal'. He's got a Raspberry Pi RP2040 board inside in the form of a Pimoroni Servo 2040, chosen for its 18 servo sockets, and senses the world through an ultrasonic range finder.

Walk the walk

Kevin wanted to create a small bipedal robot so he could develop his expertise with walking and gestures. He relished the idea of designing something that would quite literally stand or



▲ Chip shows off his many moves

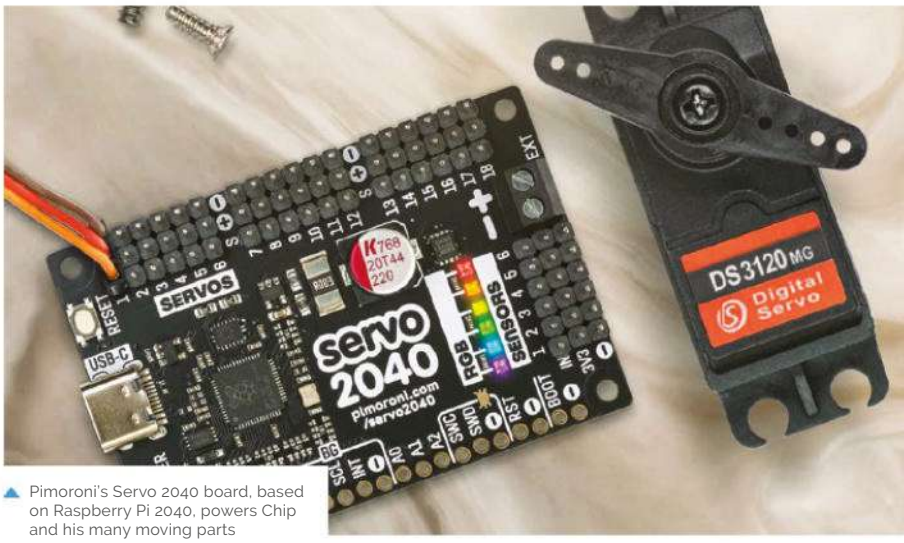

The head holds an ultrasonic range finder sensor, and was the only 3D-printed part that came out right the first time!
Chip is a Raspberry Pi RP2040-based robot that can walk, play footie, pick up objects, or simply sit on maker Kev's shoulder watching the world
He has a great range of movement thanks to the 19 servos within his limbs

Quick **FACTS**

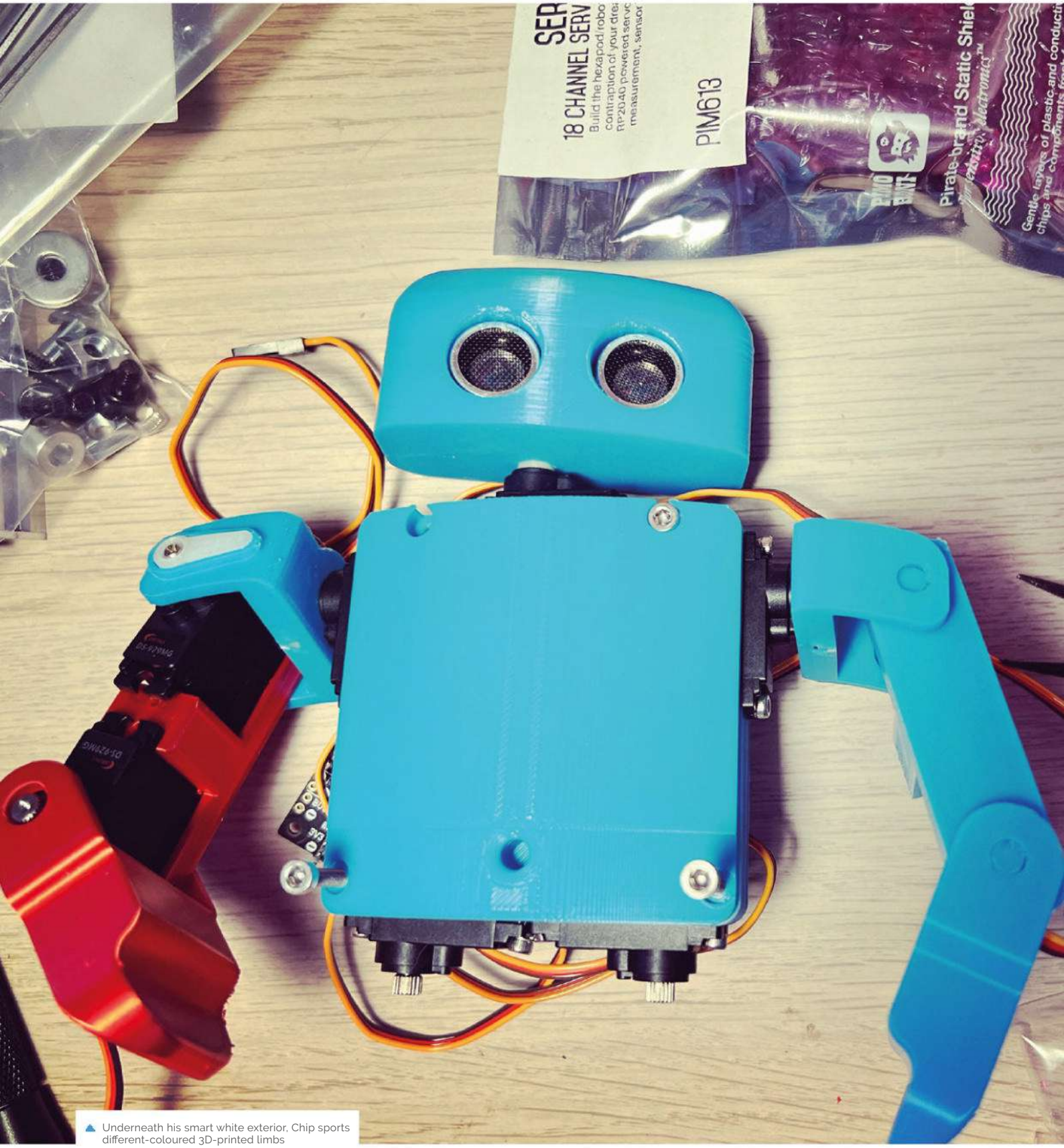
- Chip may sit on a shoulder but the pun was accidental...
- ...One Kevin immediately embraced
- Due to printing variations, Chip's legs don't match...
- ...just like his hero, C-3PO!
- Kevin is already planning a companion robot for Chip

fall depending on whether he was able to get the balance right, but also the instability needed to allow his electronic biped to walk. He envisaged a robot similar in looks to Plen (the first open-source, 3D-printable robot), with nods to Robosen (which makes the likes of Optimus Prime and Buzz Lightyear robots) and EZ-Robot, a self-assembly robot for educational use. Walking and smoothly extending the limbs were key, and meant Kevin would need to consider the robot's aesthetic design and be vigilant about 3D-printing tolerances.

The ambitious robot design will provide "18 degrees of freedom" proclaimed Kevin part-way through the build, "and move in 18 different ways." The hands, arms and shoulders, hips, upper leg, knee, lower leg, shin, and foot will all articulate, while the neck will be able to rotate. All of this necessitates a board with 18 servo sockets, hence Kevin's choice of Pimoroni's Servo 2040 board based on the RP2040 chip.



▲ Pimoroni's Servo 2040 board, based on Raspberry Pi 2040, powers Chip and his many moving parts



▲ Underneath his smart white exterior, Chip sports different-coloured 3D-printed limbs

A clear advantage for Kevin was being able to program everything in MicroPython, his first choice of programming language due to its simplicity. The straightforward code is pretty important as Kevin aims to put together a tutorial for Chip which all ages will be able to follow. He also rates Pimoroni's documentation and support, and says "Pimoroni includes some great servo classes that have servo easing to make motion really buttery smooth." Even so, he needed to adjust the tolerance (distance between the two arms between each servo) so the servos would move freely.

Rock solid foundations

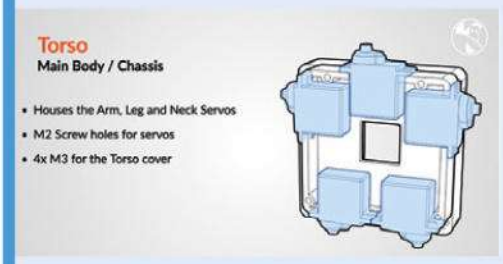
Kevin says using the RP2040-powered Pimoroni Servo 2040 board meant he could get straight to the interesting part of this project, designing the parts and constructing them. The Raspberry Pi RP2040 chip at its heart "has plenty of power and flexibility to enable me to add extra sensors, such as an IMU (inertia measurement unit), so the robot can balance and keep itself upright." However, it didn't all go quite to plan: "I burned out a couple of servos making this project; the joints were too stiff, and I cranked up the amperage to see if they could move it better, and magic smoke appeared."

> "Seeing a little human-like robot walking around is too much fun!"

Despite his experience in 3D printing and robot design, Kevin found he had to print several versions of each part to get the fit just right. "Designing the model in 3D doesn't always highlight the issues you find after 3D-printing the model," he explains. He was able to reduce the printing time as he has two 3D printers, but one uses red filament, so the pieces then didn't match, necessitating a complete body spray to give Chip the intended look.

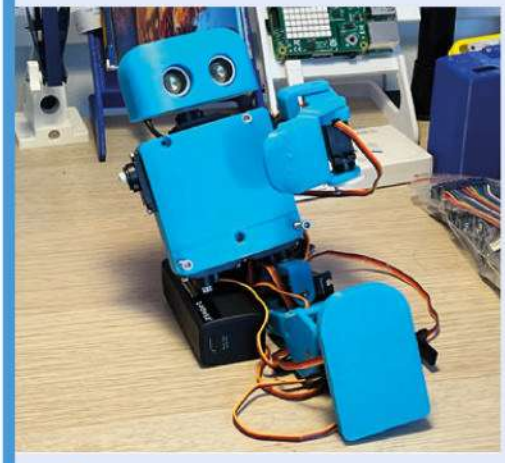
We think Chip looks great and will be keeping an eye on Kevin's blog to see how the self-balancing feature based on the PID algorithm (**magpi.cc/pidtheory**) comes along. Having splurged on 19 servos costing £9 each, Kevin admits Chip is an expensive build, but loved bringing him to life: "Seeing a little human-like robot walking around is too much fun!"

Construct a companion



01 Kevin details the build for Chip (Cute Humanoid Interactive Pal) here: **magpi.cc/chipgit**. You will need to 3D-print the parts; the STL files are at **magpi.cc/chip**.

02 The 19 servos are attached to a Pimoroni RP2040 Servo board via its headers, while attaching an IMU sensor helps keep Chip surroundings-aware and, therefore, upright.



03 The head unit's ultrasonic distance sensor doubles as eyes. The fully assembled Chip has 18 different degrees of freedom, but loves to gaze at the world from his maker's shoulder.

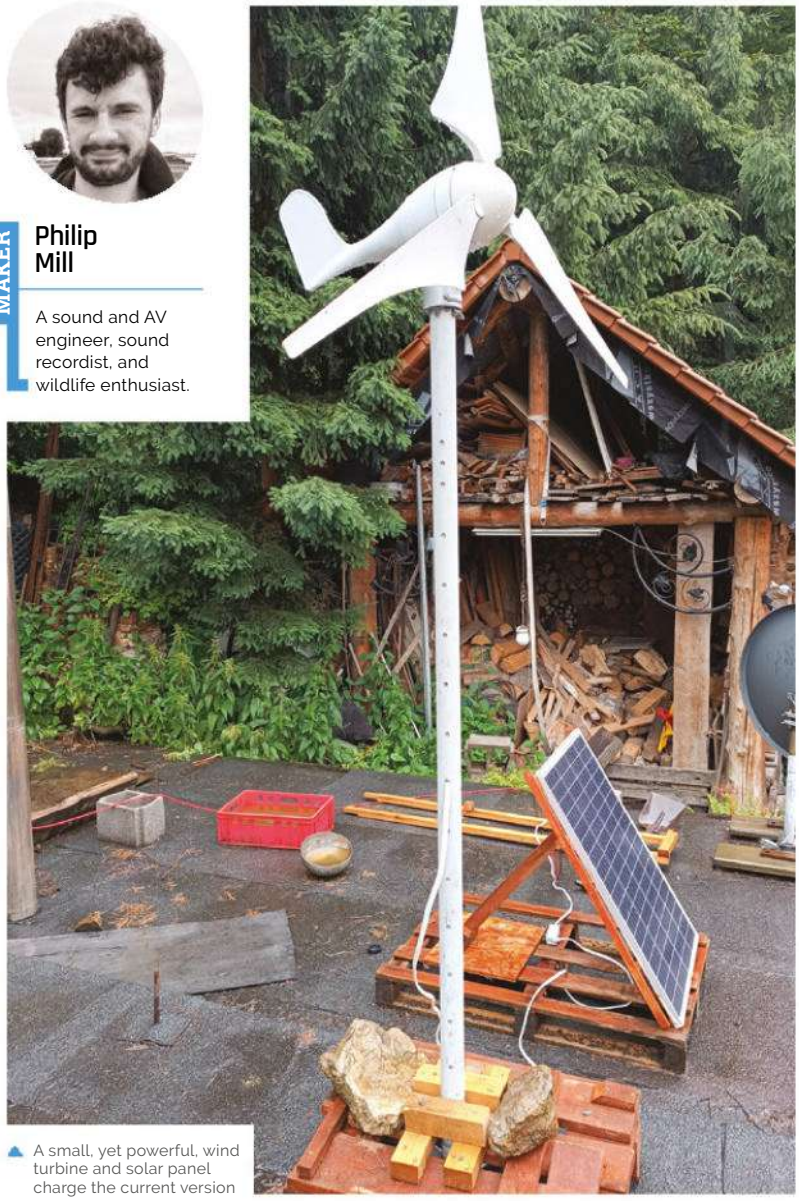
Badgercam

What do you do when you want to observe wildlife at a distance with little interference? Dig out a spare Raspberry Pi, as **Rob Zwetsloot** finds out

MAKER

Philip Mill

A sound and AV engineer, sound recordist, and wildlife enthusiast.



▲ A small, yet powerful, wind turbine and solar panel charge the current version

Warning! Observing Animals

Do not bother animals in the wild more than you have to. Any observation posts should not disturb wildlife.

It seems like we've been covering wildlife camera traps a lot recently but, to be fair, they're all very cool. Philip Mill, a sound engineer with a love of wildlife, actually got in contact with us at the start of 2023 about his current project. Perhaps it's inspired the last few months of articles?

"Originally the project was created – and I generally referred to it as – 'Badgercam', but the current iteration is not at the badger location, so now it's just Solar and Wind-powered sound recorder (with image capture)," Philip tells us. We think Badgercam rolls off the tongue a bit better though. "[This was] to address the problem and help with identifying the species or source of sounds; making sure it was the badger I recorded; and also to keep my footprint at the setts low by having a discreet recorder, and a way to see into the life of badgers without walking through their environment. They are very sensitive animals and I wanted to only visit once, or a limited amount of times!"

While he initially wanted to use a Raspberry Pi Zero, stock shortages last year resulted in him using a spare Raspberry Pi 3B+. "Raspberry Pi… had many appealing factors, including size, adaptability, and the option to connect it to a network for remote access via VNC," Philip explains. "And the [ability to add] cameras, PIR sensors, and USB inputs or an audio HAT made Raspberry Pi a logical choice. I always wanted to make a project on one and I thought what better application than this: to watch and observe badgers from my living room!"

Better camera trap

Raspberry Pi makes things easier, especially when you have an idea in mind.

However, there can always be a few snags with the parts surrounding it. "The building process, in terms of the hardware, has been a long and

Remote energy sources, such as solar and wind, can be used to keep the system operational

An IP camera takes photos, with a PIR motion sensor being used to activate it

It's important to make the smallest impact on the badgers' environment, so camouflaging is essential

frustrating (but fun!) learning process," Philip admits. "I've tried many, many combinations of different hardware, Python libraries, cameras, and housing designs to get to the point I am at now. This project is constantly growing and, despite all

> This project is constantly growing and, despite all of the challenges, I look forward to the next ones!

of the challenges, I look forward to the next ones!" The process has involved a lot of different box sizes, batteries, and solar panels, partly designed around the original idea of using a Raspberry Pi Zero.

The current version uses an 180W solar panel and a 400W wind turbine charging a 12V car

Quick FACTS

- The audio equipment used is a Zoom H1 Sound Record and Micbooster Clippy Stereo Mics
- A Tapo IP Camera C310 was the original camera used
- Philip's Dad, Robert, helped him with the coding
- The spare Raspberry Pi came from a project Philip funded
- Setts can have hundreds of metres of tunnels, and house 15 or more badgers

Philip went through many variations of the build

A camera and stereo mics record the environment

While maybe not his intended target, at least Philip knows his system works

battery to power the whole system, which has some serious sound equipment added, along with an IP camera and a wireless router. This is clearly a bit more than the few amps that a 5V Raspberry Pi usually needs.

Witness the badger

So far, all the frustrations have been worth it.

"It works fairly well; it's come along way and I am really happy with the results right now; but there is much room for improvement," Philip tells us. "Also I am still learning things about efficiency. For example, the recent challenges have been designs for the housing that holds most of the recording equipment, Wi-Fi, Raspberry Pi, etc. At one point the box got to a temperature of 64°C, and although Raspberry Pi keeps working OK, either with a fan connected to the board or a heat sink, the other parts such as the Wi-Fi stop working at such temperatures and the unit switches off. Although this isn't such an issue as it doesn't affect the recording process."

The next iteration is already being planned along with some fixes, such as RF interference from the audio recording equipment, and replacing the power-hungry IP camera.

"Recently, and I am very happy about this, I finally got my hands on a Raspberry Pi Zero [with a Raspberry Pi Camera Module NoIR attached]," Philip says. "This was the original idea for the project. Now, I am looking at options for audio and have a few HATs and some ideas for other ways to get better-quality audio. The code will most likely stay the same, but I am also considering the option of removing the PIR sensor and having the image capture by detecting motion changes based on a still image, which then will trigger the recording process. I'm excited for the next step and eventually deploying this at the badger sett; fingers crossed!"

Setting up Badgercam

01 Find a safe spot for the device, and make sure it doesn't impact on the badgers in the wild. Cameras and microphones are good passive sensors for this too.

02 Badgers are nocturnal, so you'll have to be very patient and wait until night to see some badgers live. Sometimes, waiting until the following morning works best.

03 Not quite a badger, but a night creature nonetheless. Tweak settings as you go to make sure you get stoatin' photos.

SUBSCRIBE TODAY FOR JUST £10

Get 3 issues + FREE Pico W

Subscriber Benefits

- **FREE Delivery**
  Get it fast and for FREE
- **Exclusive Offers**
  Great gifts, offers, and discounts
- **Great Savings**
  Save up to 23% compared to stores

Subscribe for £10

- Free Pico W
- 3 issues of The MagPi
- Free delivery to your door
- £10 (UK only)

Subscribe for 6 Months

- Free Pico W
- 6 issues of The MagPi
- Free delivery to your door

£30 (UK) $43 (USA)
€43 (EU) £45 (Rest of World)

Subscribe by phone: **01293 312193**

Subscribe online: **magpi.cc/subscribe**

Email: **magpi@subscriptionhelpline.co.uk**

Subscribe for £10 is a UK-only offer. The subscription will renew at £15 every three months unless cancelled. A free Pico W is included with a 6-month subscription in USA, Europe and Rest of World.

SUBSCRIBE TODAY AND GET A

FREE Raspberry Pi Pico W

Subscribe in print today and get a **FREE development board**

WORTH **$6**

- A brand new RP2040-based Raspberry Pi Pico W development board
- Learn to code with electronics and build your own projects
- Make your own home automation projects, handheld consoles, tiny robots, and much, much more

This is a limited offer. Not included with renewals. Offer subject to change or withdrawal at any time.

Buy now: magpi.cc/subscribe

SUBSCRIBE on app stores

From **£2.29**

Available on the App Store

GET IT ON Google Play

SUCCESS STORY magpi.cc/success

University of Leeds glacier research

Flexible, easy-to-use Raspberry Pi cameras and computers lower the cost of 3D glacier modelling. By **Rosie Hattersley**

After a successful European trial, researchers at the University of Leeds School of Geography (magpi.cc/uolgeog) are using Raspberry Pi computers and Raspberry Pi High Quality Cameras in low-cost, long-term glacier degradation warning systems in the Andes and Himalayas.

Among the most visible markers of the pace of climate change are the accelerating rates at which ice fields are receding and glaciers melting. NASA's climate observations confirm that the added water from glaciers and ice sheets as they melt, and the expansion of seawater as it warms, are the largest contributors to rising sea levels.

Accordingly, several important studies are closely monitoring the rate at which polar regions are changing. They include Arribada and Penguin Watch's studies using Raspberry Pi cameras for long-term time-lapse footage of specific locales, measuring changes in ice and iceberg distribution, as well as in wildlife populations over the seasons. However, there are fewer studies into continental glacier shrinkage, not least because of where such glaciers are located. One group of scientists doing this work is the team behind a glacier calving research project at Leeds University's School of Geography, led by lecturer Liam Taylor.

THE CHALLENGE

To effectively monitor glacier changes, detailed on-the-ground investigations need to go hand in hand with larger-scale observations of glacial landscapes and the environmental changes that are taking place. This is where 3D scanning and timelapse photography can play a vital role. However, while robust timelapse cameras and environmental

monitoring systems do exist, their high cost is prohibitive, since resources are often scarce in the very places that are subject to the most extreme effects of global warming. As Taylor observes, "In fact, most glaciers are located in countries like Peru and Nepal, and India."

Often, people living in these areas are at risk of food insecurity, as well as the sudden loss of their homes should an environmental catastrophe such as a flood occur as a result of glacier calving. "We are working with indigenous communities out there who are living hundreds of metres from the real risk for things like outburst floods," Taylor explains.

"The crux of the project was, can we develop something that is a monitoring technique of sorts?" says Taylor. His team wanted to develop something that could act as an early warning system for a calving event, in which pieces of glacier break off, with the potential to cause widespread flooding. The Leeds team chose to focus on glaciers because "they are receding faster than ever, the hazards associated with them are growing, and there are more and more people around the world vulnerable to hazards."

Keeping costs very low – under £100 – was an overriding principle of the Leeds team's project, given that one of its aims was to support communities in developing countries.

THE SOLUTION

The Leeds University team, Liam Taylor and colleagues Duncan Quincey and Mark Smith, completed a trial process in Iceland. They set up ten monitoring units, each built around a Raspberry Pi High Quality Camera and a Raspberry Pi Zero W, and used them to create photogrammetric images of the Fjallsjökull Glacier. The cameras were placed inside a weatherproof housing – the most expensive part of each camera setup – and were programmed to take a snapshot at preset intervals over several months. The units were powered by batteries that were replenished with solar power, while the cameras themselves were positioned to capture a linear, but overlapping, view of the glacier.

The resulting photos were collected a few months later and then processed using Agisoft Metashape software to create realistic 3D scenes. The Leeds University geographers ran concurrent tests with an unmanned aerial device (UAV, or drone) focusing on the same area of the glacier, and processed those results with Metashape too. The team compared subsets of the results from each monitoring system, consisting of images of 250-metre-long regions of the glacier.

> We're trying to go out there to get more money to put in the hands of communities so that they can spend it on these Raspberry Pi tools that we now know work

Following this successful trial, they installed Raspberry Pi-based monitoring devices in the Cordia Vilcanota, approximately 50 miles from

Cusco, Peru, where there is a series of glaciers. And over 10,000 miles away, in India and Nepal, the Leeds team has supported local researchers to set up new investigations into Himalayan glacier calving. The local teams are running the research themselves, assisted by the technical knowledge that the Leeds geographers have acquired.

The team is keen for their work to benefit as many people as possible. "The system will always be open source, and we'll always be looking to offer help and support to any citizen scientists looking to implement similar systems to respond to their own challenges."

WHY RASPBERRY PI?

The team's requirements were straightforward: they sought low-cost hardware that was capable of the task, available to buy in quantity from local resellers, and simple enough to be set up by someone with only a couple of hours of training.

Liam and colleagues looked at what was available in street markets near regions where they hoped to set up glacier monitoring. They found that Raspberry Pi computers and cameras met all of their requirements, and were available to buy locally at just a fraction of the cost of conventional hardware.

THE RESULTS

The stunning results from the cameras, even judged alongside those from drones with powerful cameras designed for surveillance tasks, exceeded everyone's expectations. Raspberry Pi cameras were able to capture 3D glacier images of similar quality to drones, at a dramatically lower cost. Better yet, since the computational demands of the task are moderate, the system also works on the super-low-cost Raspberry Pi Zero.

The team's research paper (**magpi.cc/glaciercarving**) evaluating the suitability of Raspberry Pi hardware for glacier monitoring, published in early 2023 and available for free, presents their full findings and analysis.

The Raspberry Pi camera setups can be left in place for months at a time, allowing for low-cost timelapse photogrammetry that produces 3D modelling with error tolerances well within range of the accuracy of a UAV: impressively, their mean error is less than one metre from a distance of 1.5km, with as few as ten such cameras needed to create this level of three-dimensional detail. Unlike a drone, which is subject to the vagaries of poor weather and can only fly when the conditions are favourable, the resilient Raspberry Pi camera setups perform in all weather, and require human intervention only when a researcher comes to collect an SD storage card so the photos can be analysed.

The affordability and availability of the Raspberry Pi hardware mean that local researchers can speedily assemble additional units once initial training has been completed. The Leeds University team is particularly keen for the research to be left in the hands of local environmental research teams, giving them relevant skills and knowledge along with agency over their own environmental investigations.

Work to further extend the capabilities of the monitoring setup continues: in Nepal, one of the Raspberry Pis will be used in tandem with a Raspberry Shake (**magpi.cc/shake**) seismograph, along with a microphone to listen to the glacier as it "groans along". The Leeds team also hopes to locate sensors covering a region of the Swiss-Italian Alps at high risk of glacier collapse events, with the aim of providing advance warning and protecting life.

At the end of 2022, the team headed back to Peru for an international climate summit to demonstrate their impressive photogrammetric research, and to show national and local governments that this kind of monitoring could form part of their adaptation to climate change. Their effective and affordable Raspberry Pi-based setup attracted interest from a number of government bodies in Peru, as well as charity organisations. "We're trying to go out there to get more money to put in the hands of communities so that they can spend it on these Raspberry Pi tools that we now know work."

All images © Liam Taylor, Leeds University School of Geography, reproduced by kind permission.

Astradis Elektronik is Raspberry Pi's distributor for the industry

Discover the extensive possibilities of Raspberry Pi and revolutionise your projects, industrial applications and embedded systems. With its compactness, versatility and compatibility, Raspberry Pi offers a future-proof solution. You get robust computing power and extensive peripheral support, as well as unparalleled hardware and software support trough an engaged community worldwide.

With us, you can buy Raspberry Pi's innovative products and rely on reliable and timely deliveries. Our many years of experience in logistics and warehousing enable us to offer you many ways to optimise your production process and thus reduce costs.

Time to Technology

- *In addition to Raspberry Pi products , **we offer** you many suitable components from leading manufacturers, **we optimise** your parts list and combine components into assemblies, which we then manufacture and deliver for you.*
- *In addition, **we provide** you with development tools for Raspberry Pi products that we have created.*

  These include, for example, HATs for mobile communication, displays that plug- and-play Raspberry Pi drivers, and development boards with a variety of interfaces to exploit the full potential of Raspberry Pi.

ATXX 5G Shield HAT

Time to Market

- *To successfully bring your product to market, **we help you optimise the design** and **find the best solution for distribution**.*
- ***We offer flexible external storage solutions** from micro SD cards to integrated SSD hard drives.*
- *Together with our partners, **we can securely flash your software and meet your security requirements**.*
- ***We are happy to organise production for you** and **deliver a finished ODM product with the „Powered by Raspberry Pi" label**.*

Do you have an idea for partnership? Please contact us @ a.feix@atxx.de; Phone: +49 89 89 505-231

Emulate EVERY THING

Use ready-made emulation distros to turn Raspberry Pi into an all-in-one emulator to play the best classic and modern retro games

BY KG ORPHANIDES

Raspberry Pi can be fully transformed into a DIY multi-console, thanks to three polished emulation-oriented operating systems: RetroPie, Recalbox, and Lakka. They're all excellent and include a wide range of emulators, a notable number of which are maintained by the Libretro project.

All three are available as image files that you can both download and write to an SD card using Raspberry Pi Imager (**magpi.cc/imager**). Once you've connected them to your network, you can just browse to their SMB shared directories from any computer, and drop your ROM and BIOS files over to play.

CHOOSING YOUR PERFECT DISTRO

All three retro gaming distributions support a wide range of controllers, including USB and Bluetooth console pads, retro joysticks, and full arcade setups connected via USB or GPIO. You can use a keyboard as well, although this is not the best way to navigate their menus and play games, so we strongly recommend investing in a joypad. The Sony DualShock 4 is a reliable choice.

Recalbox has also recently started manufacturing hardware, such as a SCART adapter for Raspberry Pi, making it significantly easier to connect your emulation console to an old-school CRT TV with the Recalbox RGB Dual (**magpi.cc/rgbdua**).

Both Recalbox and RetroPie use the EmulationStation front-end, customised to provide characteristic user experiences, and

LAKKA

Developed by the Libretro team behind the RetroArch front-end and libretro library of emulator cores, Lakka is a lightweight emulation distro based on LibreELEC. While its interface doesn't feel as polished or attractive as its rivals', Lakka gets the latest stable libretro emulator updates first and releases special versions for a range of emulation hardware, such as DIY handhelds. The other emulation distros we've looked at are often partly built upon Libretro's work. However, while Libretro's arcade and console emulators are among the best around, its vintage microcomputer emulators leave something to be desired, particularly when it comes to keyboard input handling.

lakka.tv

Warning! Copyright material

This feature discusses the use of legally distributed games and system ROM images for a variety of vintage computers. However, please note that many other titles are distributed online without their rights holders' permission, and are not legal to use under UK law.

magpi.cc/legalroms

with a range of additional supported themes. If you're building an arcade machine, Recalbox's large, friendly default interface and browser-accessible configuration and game ROM upload tools, alongside SMB shares, make it particularly appropriate for stand-alone cabinets. Recalbox also provides customised support for a number of Raspberry Pi-powered handhelds.

However, Recalbox has a fixed, somewhat limited selection of emulators and versions. These are tried and tested, but it means edge cases won't be catered to when it comes to game support.

Meanwhile, RetroPie's more intimate interface, smaller fonts, and in-depth, exclusively keyboard-navigated emulator module configuration make it a good call for multi-console or even computer emulation, particularly if you've got a keyboard immediately to hand.

RetroPie, built on top of Raspberry Pi OS, is also more customisable and provides a wider range of emulators, including dozens of experimental and niche options. However, unlike Recalbox and Lakka, a 64-bit version of RetroPie isn't currently available. This is largely inconsequential, but a handful of emulators have better performance in their 64-bit versions

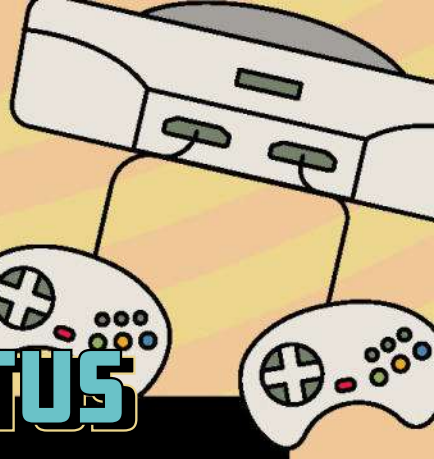
Finally, Lakka is the official distro of the RetroArch front-end and libretro emulator ecosystem. Its interface is more workmanlike and bare-bones than its counterparts. This makes Lakka less pick-up-and-play friendly than either RetroPie or Recalbox. However, Lakka makes it very easy to try different emulators on the same game to find the one that works best for it, which has advantages.

Lakka makes it very easy to try different emulators on the same game

RECALBOX

Recalbox's attractive and easy to navigate EmulationStation-based front-end design works particularly well on handhelds and arcade machines, and custom versions are available for a range of retro console hardware kits, many of which are autodetected in order to load appropriate controller configurations. Recalbox also makes the RGB Dual HAT, custom hardware that adds high-quality CRT display output to Raspberry Pi, and that's only fully supported in this distro, so far. The disadvantage is that Recalbox ships with a fixed range of emulators that you can't add to, upgrade, or downgrade independently of the OS. However, scores of different emulators are included, including non-default alternatives that you can manually select to try specific games with.

recalbox.com



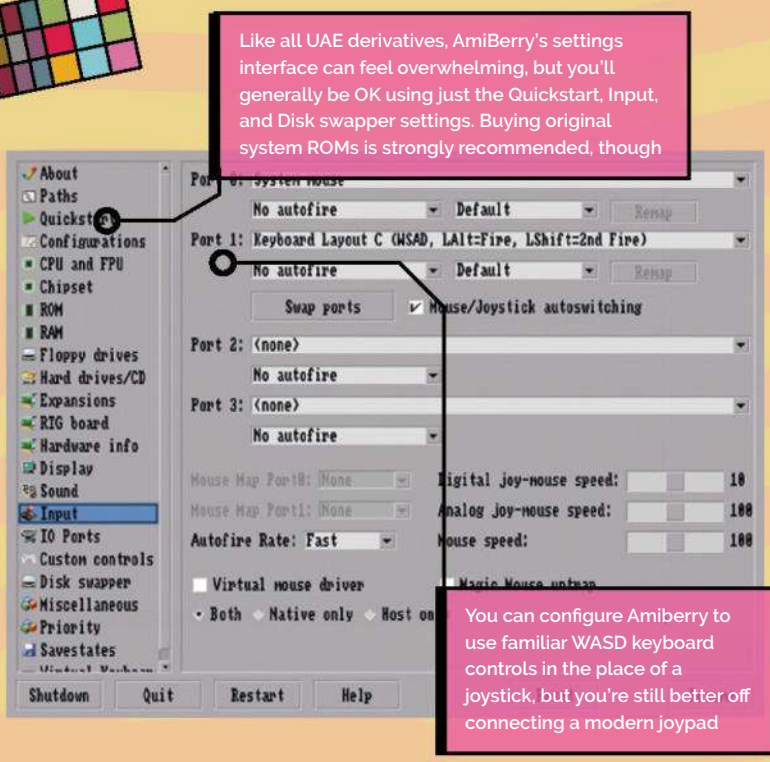
BIOS & ROM LEGAL STATUS

Some emulators require BIOS ROM images taken from the original hardware. These can sometimes be used legally in the UK, because permission has specifically been given by the rights holder to either copy the BIOS onto a modern PC, or download the BIOS ROM from the internet. This table shows all systems that we know either don't require BIOS images, or for which appropriate images are legally available.

PLATFORM	ADDITIONAL BIOS OR FIRMWARE IMAGES REQUIRED?	LEGAL IMAGES AVAILABLE FOR FREE?	SOURCE	LEGAL IMAGES AVAILABLE TO BUY?	SOURCE
3DO	Yes	No	N/A	No	N/A
Amstrad microcomputers	Yes	Yes (since 1999)	magpi.cc/amstradrompermission	No	N/A
Atari Jaguar	Only required for some games. See magpi.cc/virtualjaguar	No	N/A	No	N/A
Atari ST	Yes	Yes	magpi.cc/emutos	No	N/A
Commodore Amiga	Yes	Yes	aros.org	Yes	amigaforever.com
Commodore C64	Yes	Yes	magpi.cc/c64foreverexpress, magpi.cc/mega65openroms	Yes	magpi.cc/c64foreverplus
DOS	No	N/A	N/A	N/A	N/A
MSX microcomputers	Yes	Yes	cbios.sourceforge.net	No	
Nintendo 64	Optional	No	N/A	No	N/A
Nintendo Entertainment System	No	No	N/A	No	N/A
Nintendo Game Boy	No	No	N/A	No	N/A
Nintendo Game Boy Advance	Optional – required for Joybus link functionality	Yes	HLE bios included in emulators	No	N/A
Nintendo Game Boy Color	No	No	N/A	No	N/A
Nintendo GameCube	Optional – required for some font support	Dolphin HLE BIOS	magpi.cc/dolphinemu	No	N/A
Nintendo Super Nintendo Entertainment System	No	N/A	N/A	No	N/A
Sega Game Gear	No	N/A	N/A	No	N/A
Sega Master System	No	N/A	N/A	No	N/A
Sega Mega Drive	No	N/A	N/A	No	N/A
Sega Dreamcast	Yes	Yes	redream.io	Yes	Paid-for version
SNK Neo-Geo	Yes	No	N/A	Yes	DotEmu SNK 30th Anniversary edition
Sony Playstation	Yes	PCSX-Reloaded HLE BIOS	magpi.cc/pcsx Included in PCSX-Reloaded and PCSX-Rearmed emulators	No	N/A
ZX Spectrum	Yes	Yes (since 1999)	magpi.cc/amstradrompermission	No	N/A

Emulate on the DESKTOP

Build an old-school micro or desktop PC gaming experience on top of Raspberry Pi OS



While having a DIY games console, handheld, or arcade machine is incredibly cool, it's not always the best setup for playing old school computer games, especially those that require a keyboard and mouse. For this, a proper desktop environment really comes into its own, although you may still want to reduce your resolution to optimise the performance and appearance of older titles.

DOS and early Windows emulation are probably the most popular and accessible options, but Microsoft operating systems and Intel hardware aren't the only computer platforms that lend themselves to easy desktop emulation. We'll also run C64, Commodore Amiga, and Atari ST games.

01 GET READY TO PLAY

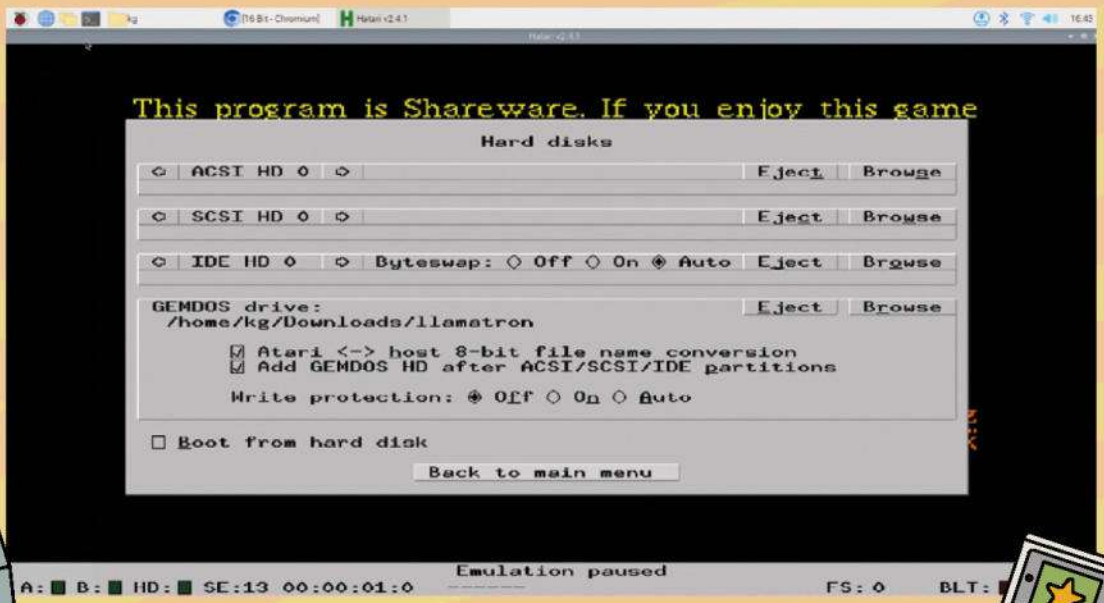
Create directories in your home directory called **Games** and **Software**. Some of our emulators will live in **Software**, but let's grab some games first. Create subdirectories for each system you're going to emulate to house the games we'll be playing: **atari**, **amiga**, **c64**, and **dos**. After downloading them, extract each of the compressed game files into an appropriately named subdirectory of those folders.

02 DOWNLOAD FREE GAMES

A number of developers have made freeware versions of their games available at **scummvm.org/games**. There are plenty to choose from, but we'll use the CD version of Revolution Software's Beneath a Steel Sky and extract it into **~/Games/dos/BASS**. Sticking with the classics, grab the shareware version of Doom 1.9 for DOS from **magpi.cc/doom19** and unzip it into **~/Games/dos/doom**.

YOU'LL NEED

> Raspberry Pi
> Raspberry Pi 64-bit OS
> Optional joystick or joypad (recommended for ST and Amiga games)



Hatari is one of the most faithful ST/STe emulators around, but you can still just mount a local directory full of PRG files as a virtual hard disk. It's not immediately obvious

On the Atari ST front, Jeff Minter's glorious Llamatron is available at **minotaurproject.co.uk/lc-16bit**. For the Amiga, you can download Sword of Sodan from its creator's site at **magpi.cc/sodan**. Our C64 game is Nixy the Glade Sprite, available from **magpi.cc/nixy** on a pay-what-you-want basis, starting at free – create a subdirectory called **nixy** under **c64**.

03 INSTALL THE LATEST VERSION OF SCUMMVM

To get the latest version of ScummVM with support for the widest range of games, go to **scummvm.org/downloads** and download the appropriate 32-bit (armhf) or 64-bit (aarch64) AppImage for Raspberry Pi OS. At the time of writing, the current release version is 2.7.0. Put the file wherever you like – perhaps in a **Software** directory. Right-click it, select Properties, go to the Permissions tab, open the Execute pulldown and select Only Owner, then click OK. Double-click on the file and select Execute when prompted. Assuming you want useful shortcuts, click Yes when a prompt asks you if you've like to integrate ScummVM with your system.

ScummVM makes it easy to experience and save-scum around the regular deaths that embody vintage adventure games

04 PLAY ADVENTURE GAMES

Once installed, you'll find ScummVM in the Games section of the Raspberry Pi OS desktop menu after this. To add a game, click Add Game..., browse to the directory you unzipped Beneath a Steel Sky into, and select Choose before following the prompts. You can also use the 'Mass add' option available as a pull-down to point ScummVM at your games directory.

05 ADD FLATPAK SUPPORT

The easiest way to install and update the 64-bit version of DOSBox-X for Raspberry Pi is via Flatpak, so we'll start by adding support for this convenient package management system, which bundles up all of a program's dependencies. Open a terminal and type:

```
sudo apt install flatpak
```

Log out of your desktop session and log back in again to ensure that Flatpak's environment variables are now in the path. Open a terminal and type:

```
flatpak remote-add --if-not-exists flathub
https://flathub.org/repo/flathub.flatpakrepo
```

06 INSTALL DOSBOX-X AND DOOM

In a terminal, type:

```
flatpak install dosbox-x
```

Say Yes to all options to install it. Reboot and you'll find DOSBox-X in your Games menu. The first time you run DOSBox-X, you'll be prompted to select a working directory. Select the **~/Games/dos** directory we made earlier and click OK. At DOSBox's command prompt, type:

```
mount c ~Games/dos
c:
cd doom
install
```

If you follow the suggested defaults, once installation is complete, you'll be prompted to launch the game, which will be installed to **~/Games/dos/DOOMS**.

07 INSTALL HATARI

Like DOSBox-X, the best way to keep up-to-date with the Hatari Atari ST and STE emulator on Raspberry Pi running a 64-bit operating system is to install it as a Flatpak. Open a terminal and type:

```
flatpak install hatari
```

Then answer **Y** to all prompts to complete installation. As Flatpak is depreciating 32-bit packages, those running 32-bit operating systems on Raspberry Pi will have to use the version in the OS's Apt repository. This is a little older but should still do everything you need it to. Open a terminal and type:

```
sudo apt install hatari
```

Press **F12** (**Super** + **F2** on a standard Raspberry Pi keyboard) to access settings.

08 PLAY ST GAMES

Hatari can be a little awkward when it comes to browsing your entire home directory. The Flatpak installation will set your home directory as its default path, but during our last test, only files and subdirectories in our **Downloads** directory were shown, and you can't type in the path of your choice. If you experience this issue, place the directories containing your Atari ST games into **Downloads**.

On the Atari ST, Jeff Minter's glorious Llamatron is available at the Minotaur Project

If you've got ST files, these are usually floppy disk images, and should be mounted from floppy disks. If you've got PRG executables, like our copy of Llamatron, you'll need to mount it as a virtual hard disk. Open settings, click the Hard disks button and, under GEMDOS drive, browse to the folder containing your PRG files and select it. Tick both the 'Atari <> host 8-bit file name conversion' and 'Add GEMDOS HD after ASCI/SCSI/IDE partitions' boxes, then go back to the main menu.

Click 'Joysticks setup' to check your joypad configuration. If you've not got one, select 'use keyboard' - this defaults to the arrow keys to move and right control to dire, but if you prefer WASD, just click 'Define keys'. Go back to the main menu, select 'Reset machine' and click OK.

While many developers have sold rights to their games to re-release houses, Llamasoft has made multi-platform classics free to download

QUICK TIP

Courier DOSBox-X

Flatpak images are 64-bit only. If you're running a 32-bit distro, refer to our guide at **magpi.cc/builddos**.

DOWNLOAD THE FULL CODE:
magpi.cc/viceromcopier

▲ Vice's loading times can be authentically slow, but you can hit **ALT** + **W** to enable Warp Mode. Just remember to turn it off before you start playing

The GEM desktop should now show DISK C, which contains your programs. Double-click to open it and then run the **TRON_512.PRG** file.

09 INSTALL AMIBERRY

Download the latest version of Amiberry for your Raspberry Pi from **magpi.cc/amiberry**. Create an Amiberry subdirectory under **~/Software** and extract the compressed file into that. For the most reliable loading of games from ADF disk images, including Sword of Sodan, you should ideally buy the original Amiga Kickstart ROMs from Amiga Forever (**amigaforever.com**) and place them in **~/Software/Amiberry/kickstarts**. Now open a terminal and type:

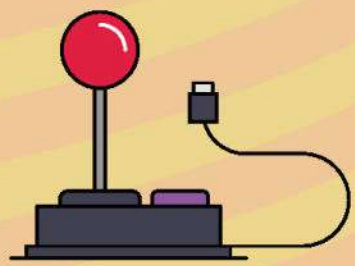
```
sudo apt install libsdl2-2.0-0 libsdl2-ttf-2.0-0 libsdl2-image-2.0-0 flac mpg123 libmpeg2-4 libserialport0
cd Software/Amiberry
```

10 PLAY SWORD OF SODAN

In a terminal, enter:

```
./amiberry
```

It'll open on the Quickstart screen. Tick the boxes next to DF2 and DF3 to enable extra virtual floppy drives. Now click on the Disk swapper tab and, for each of the first four slots, browse to one of the four ADF files (sodan1 through 4) in your extracted copy of Sword of Sodan. If there's a blank space instead of the drive number next to the file path, click on the button – these should say DF0: through DF3:. Finally, configure your controls. Go to input and, if you're not using a joystick, enable the Keyboard Layout of your choice for Port 1. Finally, click Start. When using your emulated Amiga, press **F12** at any point to open Amiberry's settings. To wield your sword, press fire at the same time as the direction in which you wish to attack.



11 INSTALL VICE

The VICE Commodore 8-bit emulator has returned to Raspberry Pi OS's repositories, now unencumbered by ROMs with potentially disputable licences. That means you'll have to get hold of your own C64 ROMs. Fortunately, they're available for free from Cloanto. This is distributed as a Windows MSI file, so we'll use the Gnome MSI Tools to extract the files it contains and we've written a script to move them to the right places.

Go to **c64forever.com** and click Download Free Express Edition. Open a terminal and type:

```
cd Downloads
sudo apt install msitools vice
mkdir C64Forever && mv C64Forever10.msi ~/Downloads/C64Forever && cd C64Forever
msiextract C64Forever10.msi
cd 'Program Files/Cloanto/C64 Forever/Shared'
wget https://codeberg.org/MightyOwlbear/VICE-Forever-ROM-copier/raw/branch/main/copyrom.sh
sudo sh copyrom.sh
```

12 RUN A C64 GAME

Make sure your controller is plugged in, then go to the terminal and type:

```
x64
```

Now you've run VICE's C64 emulator, let's configure it and give it something to emulate. Go to Settings > Settings and select Input Devices > Joystick. Assign your controller to Joystick #2.

Open the File menu and select Smart Attach Disk/Tape/Cartridge.

Browse to **~/Games/c64/nixy/prg** and select **nixy.prg**. Once it's loaded – this takes a minute or so – READY will be displayed on the emulated computer. Now type RUN.

Press **ALT** + **D** for full-screen mode. If your sound isn't working, try pressing **ALT** + **W** to disable warp mode, which gives you faster load times but automatically disables audio.

QUICK TIP

More than just point and click

ScummVM started with adventure games, but now plays many classic RPGs, too. See its full supported game list at **magpi.cc/scummvmsupport**.

Software HUNTING

The DotEmu ports of SNK's classic beat-'em-ups and shooters like Ironclad include everything you need to run them on the emulator of your choice

The MagPi magazine's guide to buying the best retro re-releases and modern classics

Retro DOS and early Windows games are both popular and easily available, thanks to official re-releases on digital store-fronts like GOG, Zoom Platform, and even Steam, which kept the games accessible, even after the original boxed releases became collectable and started commanding wildly excessive prices on eBay. Meanwhile, homebrew scenes have kept alive classic 8-bit and 16-bit consoles and computers, with varying levels of support from their original creators.

This emulation boom has in turn prompted some new publishers to buy up the rights to ZX Spectrum, C64, Amiga and Atari ST classics and sell them for anywhere from £3 to £25 a pop. Helpfully, almost all of these include original disk images that you can run on any emulation system, but watch out for remakes that only work on modern computes. But vintage games often pale in comparison to the quality and innovation of modern commercial and homebrew games.

BUYING DOS GAMES

GOG usually packages its retro games for Windows users, usually bundled with DOSBox, using InnoSetup. Using the innoextract tool from Raspberry Pi OS's repositories, we can extract those game files to use with our own emulator without having to install them on a separate PC with x64 or x86 architecture. You'll even find several vintage classics available for free at **magpi.cc/gogfree**.

BUYING RE-RELEASES ON STEAM

A number of publishers are putting out emulated retro games on Steam. These include Pixel Games (**magpi.cc/pixelgames**), SNEG (**magpi.cc/sneg**), whose releases you'll also find on GOG, the often-resurrected Apogee Entertainment (**magpi.cc/apogee**), the even more frequently resurrected Atari (**magpi.cc/atari**) – which also publishes some new releases, and Ziggurat Interactive (**magpi.cc/ziggurat**), among others.

A number of retro computer games have made their way onto Steam, such as Gremlin Graphics' sci-fi platform shooter, Venus the Flytrap

SEGA CLASSICS

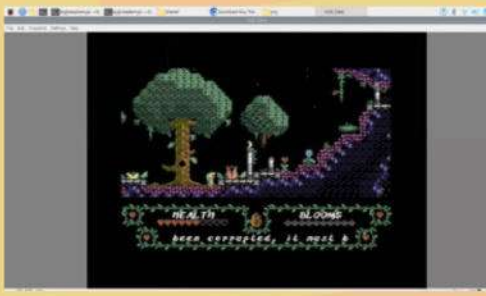
Sega is famously generous with its Mega Drive images. Just buy a copy of Sega Mega Drive & Genesis Classics on Steam (or any other digital store-front), go to the folder called 'uncompressed ROMs' and copy them over to your emulator. You may need to rename some extensions to .bin, depending on which emulator you use.

SNK

Dotemu's 30th anniversary SNK arcade re-release series just gives you the arcade ROM image for each game, plus a file called **neogeo.zip** which contains the relevant BIOS ROMs for SNK arcade machine hardware.

Note that for the latest version of the FBNeo emulator, you'll need versions of the Dotemu games, which have been recently updated with new BIOS images, such as those given away to Amazon Prime Gaming subscribers.

Note that the 40th anniversary releases produced by Digital Eclipse and released on GOG and Steam 40th are not so helpful, and instead contain everything in .mbundle files which require an inconvenient and legally dubious amount of unpacking to actually spit out the game and BIOS files.



Modern C64 games like Nixy the Glade Sprite more than stand up to what developers were creating in the 1980s

Part 01

Practical Python with Raspberry Pi

Learn to code with the Python programming language

MAKER

Lucy Hattersley

Lucy is editor of *The MagPi* and has been coding since somebody pushed her nose into a BBC Micro at school.

magpi.cc

In this, the first of a series of tutorials, we're going to look at learning to use Python to create a variety of practical projects. We'll cover all the basics, from creating your very first program to understanding how variables work, understanding flow and loops, and understanding mystifying subjects like object-orientated programming.

Along the way, we're going to try and keep things as practical as possible, exploring what you can make with Raspberry Pi and Python, from building web-based software to electronic gadgets.

To begin, we're going to write our very first program! Let's get started...

In this tutorial, we're going to use Raspberry Pi OS to practise Python. Raspberry Pi OS has Python pre-installed alongside a programming tool called Thonny.

Thonny is a type of program known as an IDE, or 'Integrated Development Environment'. IDEs make it easier to type and run programs, although you do not need one. You can use any text editor to write a Python program, and the Terminal app to run your code.

Why Raspberry Pi?

Raspberry Pi stemmed from a desire to counter a decline in computer science engagement at Cambridge University. Eben Upton, and a host of other engineers, decided that a small hackable computer, reminiscent of the computers they grew up with in the 1980s, was the very thing.

Raspberry Pi went on to capture the heart of the maker community due to its affordable and customisable nature. It was a big hit with enthusiasts because of its capacity for building and experimentation, blending hardware and software harmoniously in exciting projects.

Raspberry Pi is the ideal computer for learning to code. It is a low-cost computer that is simple to set up. Raspberry Pi runs Linux, a free operating system that's loved by coders. And you can hook up electronic components to its GPIO pins.

Start Thonny

Start Thonny by clicking on the Raspberry Pi OS applications menu and choosing Programming > Thonny. It will display two main windows: the Editor at the top and the Shell below.

It's a cultural tradition amongst programmers to output 'Hello, World!' as the first program for any new programming language.

Click on the Shell and enter the following code:

```
print("Hello, World!")
```

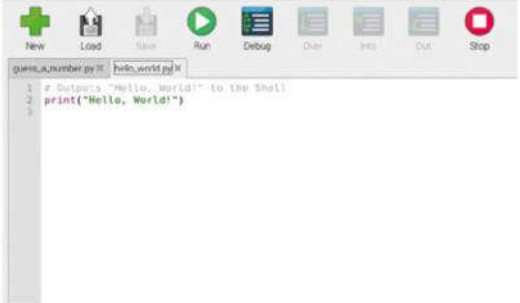
... press **RETURN** and the Shell will display:

```
Hello, World!
```

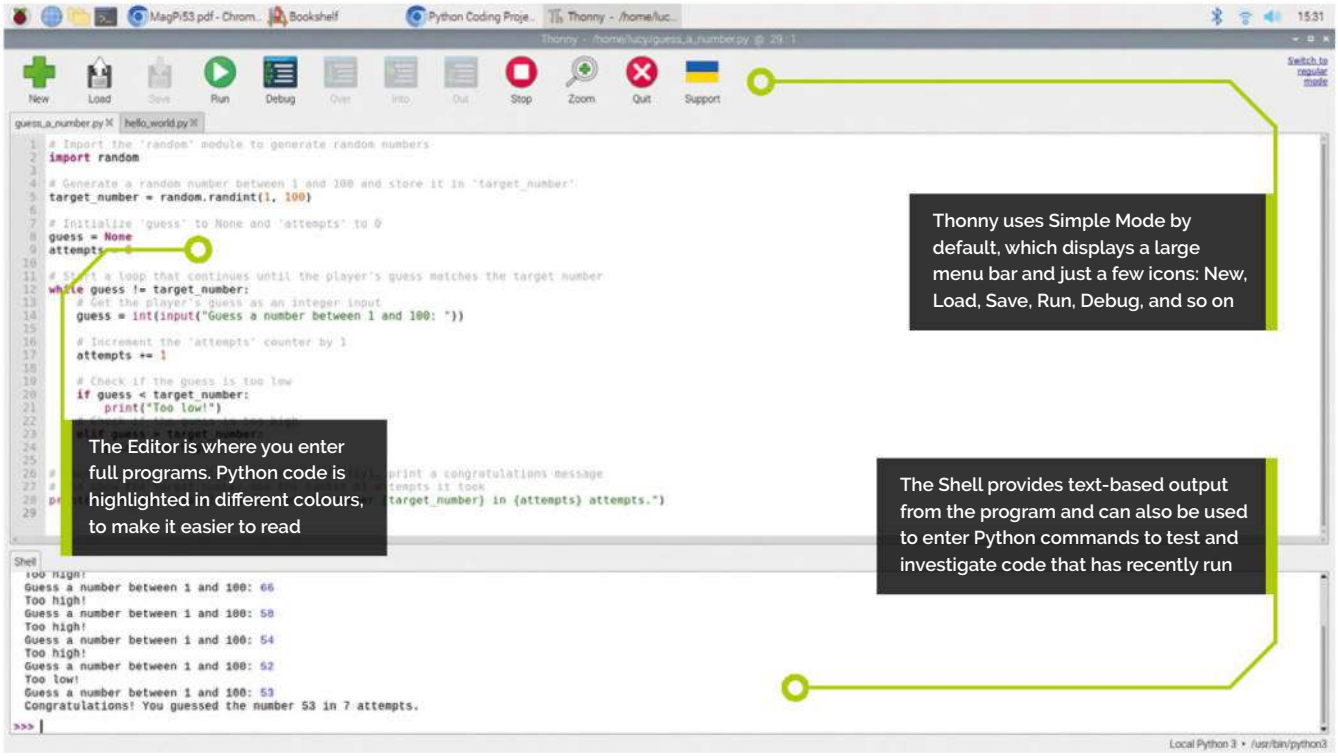
Congratulations, you've just entered your very first line of Python code.

Let's turn this into a program we can run. Click on the Editor, and enter the following code into it. Enter each character exactly as you see it, with the same capitalisation and spacing:

```
# Outputs "Hello, World!" to the Shell
print("Hello, World!")
```



Hello, World! This is the traditional first program to write in any new coding language


Thonny uses Simple Mode by default, which displays a large menu bar and just a few icons: New, Load, Save, Run, Debug, and so on
The Editor is where you enter full programs. Python code is highlighted in different colours, to make it easier to read
The Shell provides text-based output from the program and can also be used to enter Python commands to test and investigate code that has recently run

> It's a cultural tradition amongst programmers to output 'Hello, World!' as the first program

Click Save and name the file 'hello_world.py'. You can name the file anything you like, but it's good practice to describe the program. In Python, it is conventional to use lower-case letters and underscores instead of spaces. You must include '.py' as the extension though, or the program will not run.

Speaking of which! Let's get it running. Click the Run icon and 'Hello, World!' will again appear in the Shell. Click it again and it will run again.

If you see anything other than 'Hello, World!' in the Shell window then you have entered the code incorrectly. Check it carefully and take a look at the 'Debug your code' boxout for more information.

Debug your code

Most of the time your programs will not run, or run correctly, on the first pass. Do not worry. This is perfectly normal. Coding is sometimes described as the process of removing bugs (errors in the code). There are almost always bugs in everybody's code at first. Our Hello World program is pretty simple, so you will most likely run it first time. But if you were to change it to something incorrect, such as using prin instead of print, the Shell will return an error message:

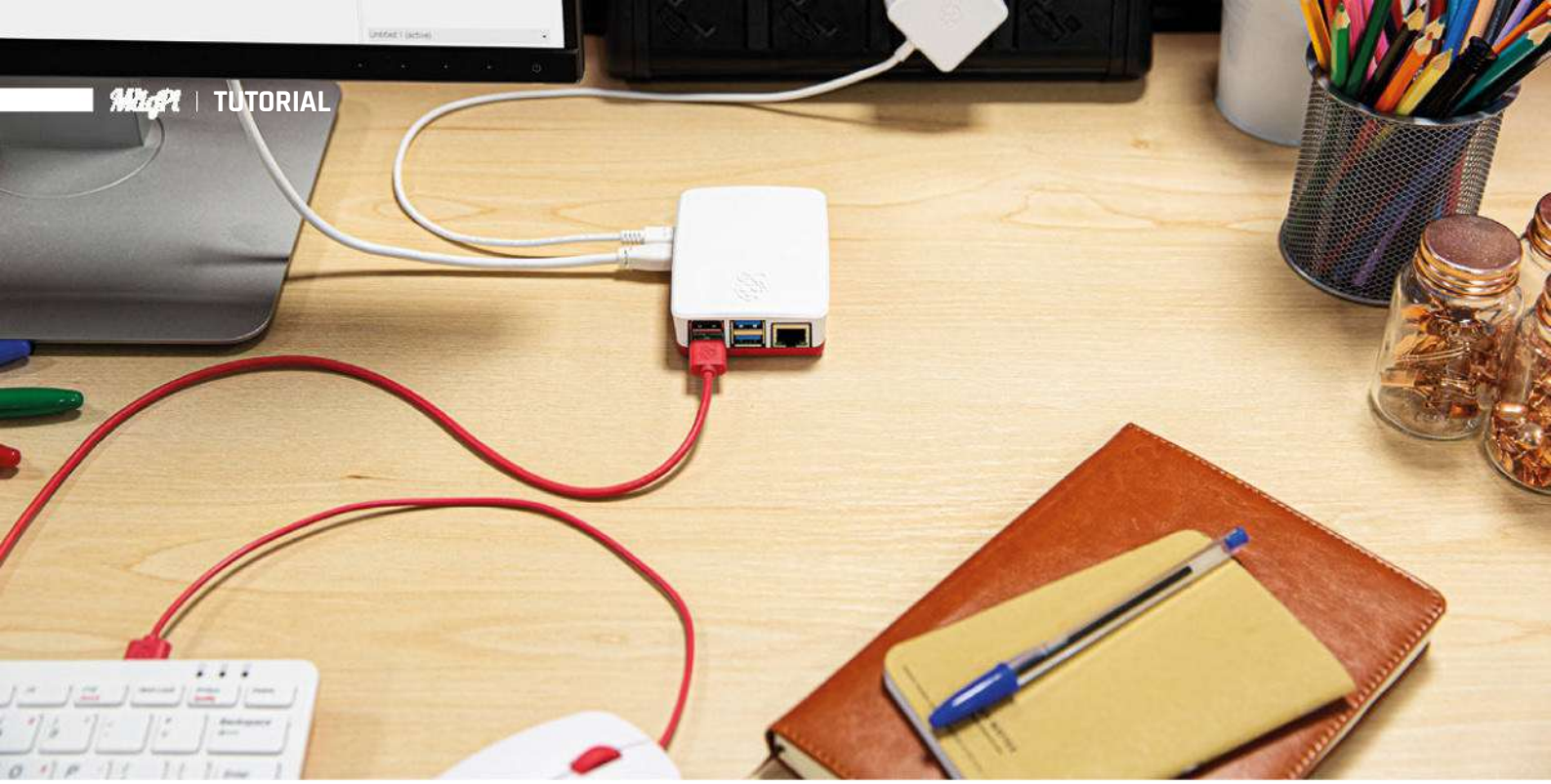
```
Traceback (most recent call last):
  File "/home/pi/hello_world.py", line 2,
in <module>
    prin("Hello, World!")
NameError: name 'prin' is not defined
```

Read the message carefully and you will get information about the error. In this case, it is on 'line 2' of our **hello_world.py** program. The line with the error is displayed and a hint to the problem: `NameError: name 'prin' is not defined`. Reading the error message carefully can help you isolate and fix errors.

Top Tip

Hello, World?

Hello, World dates all the way back to 1972 where it was used in a seminal book *The C Programming Language* by Brian Kernighan and Dennis Ritchie.



Raspberry Pi is the ideal small computer for learning to code in Python

Comments and code

Let's take a look at our **hello_world.py** code in more detail. There are two lines here. The first starts with a '#' character (an octothorpe or hash). This lets Python know that the line is a 'comment'. Comments are used to explain code to human beings and are completely ignored by Python. You can write anything in a comment. You can test this by changing or deleting Line 1 so our code just shows:

```
print("Hello, World!")
```

Click Run again and the Shell will display the same "Hello, World!" message. If you delete the `print()` code and just include the comment:

```
# Outputs "Hello, World!" to the Shell
```

Code matters

Coding is one of the most important skills you can learn in the modern age.

On a practical level, learning to code enables you to control a computer and get it to do things for you, whether it's building a website, making games, or scripting boring tasks so the computer does them for you.

The good news is that coding proficiency does not require genius, just as literacy doesn't require brilliance. The process becomes relatively straightforward once you grasp fundamental concepts like variables, branching, and loops. If you can read and write, you can code.

More importantly, learning to code teaches you critical thinking skills that help you understand how the modern world works. Code is a critical layer in our lives that sits between us and the increasingly digital world that surrounds us.

With just a small amount of understanding of how code works, you'll be able to perform computer tasks faster and get a better understanding of the world around you. Increasingly, humans and machines are working together.

"I think everybody in this country should learn to program a computer," said Apple's co-founder Steve Jobs, "because it teaches you how to think."

> More complex code is coming down the line and writing comments alongside your code is good practice

... and click Run, the Shell will display nothing. That's because, to Python, our program has no code at all.

While our Hello World code is pretty simple, more complex code is coming down the line and writing comments alongside your code is good practice, so we're going to start as we mean to go out. Writing good comments is an art form, but typically each comment should accurately describe what the code does in plain language (in our case 'Outputs "Hello, World!" to the Shell') and if there is any more complex behaviour, the comment should explain how it works.

Line 2 is our code, and this is where things get more interesting. There are three parts to this code. The first is this function:

```
print()
```

This is a built-in function for Python and it 'prints' (or outputs) whatever is in the parentheses to the Shell. You can tell it is a function because the command is followed by parentheses: '()'.

Inside the parentheses is our string of characters: "Hello, World!". The bit inside the parentheses is known as the 'argument'. Don't worry, it's not a battle, although sometimes it may feel like one.

Why Python?

Python is a great programming language and has rapidly become the de facto language for beginners and experts alike. Python code is easy to read and understand. It's powerful too, handling everything from scientific analysis to artificial intelligence. Don't be fooled into thinking Python is for newbies; Python is a serious coding language. And there is an army of experts ready to give you a helping hand. Python is the place to be.

Guess a number

Now that we have explored the basics of Python on Raspberry Pi, it's time to enter a more complete program. Click on New in Thonny and Save, enter the file name 'guess_a_number.py' and enter the code from the **guess_a_number.py** listing.

This code creates a number-guessing game. The computer picks a random number between 1 and 100 and you enter guesses until you get the number correct.

Enter the code exactly as you see it in the listing. Notice that the lines underneath the `while` code are indented by exactly four spaces (not three or five spaces or a tab – exactly four spaces).

```
 # Start a loop that continues until the
player's guess matches the target number

 while guess != target_number:
     # Get the player's guess as an integer
input
     guess = int(input("Guess a number between
1 and 100: "))
```

This is your first loop, which is a concept we will explore in later chapters. The comments give a clue as to what each part of the code does. As for now, understanding the code doesn't matter as much as practising entering a program line-by-line. Don't copy and paste the code from the digital edition or our GitHub page (although it is available for download). It's important to practise careful code entry.

After you have finished typing the code, read each line of the code carefully backwards from the last line back to the first (this helps you scan the line carefully instead of eye-skimming). Press Run, and play your first game.

hello_world.py

DOWNLOAD THE FULL CODE: magpi.cc/github

Language: **Python**

```
001. # Outputs "Hello, World!" to the Shell
002. print("Hello, World!")
```

guess_a_number.py

Language: **Python**

```
001. # Import the 'random' module to generate random numbers
002. import random
003.
004. # Generate a random number between 1 and 100 and store
     it in 'target_number'
005. target_number = random.randint(1, 100)
006.
007. # Initialize 'guess' to None and 'attempts' to 0
008. guess = None
009. attempts = 0
010.
011. # Start a loop that continues until the player's guess
     matches the target number
012. while guess != target_number:
013.     # Get the player's guess as an integer input
014.     guess = int(input("Guess a number between 1 and
     100: "))
015.
016.     # Increment the 'attempts' counter by 1
017.     attempts += 1
018.
019.     # Check if the guess is too low
020.     if guess < target_number:
021.         print("Too low!")
022.     # Check if the guess is too high
023.     elif guess > target_number:
024.         print("Too high!")
025.
026. # When the loop ends (player guessed correctly), print
     a congratulations message
027. # and show the target number and the number of attempts
     it took
028. print(f"Congratulations! You guessed the number
     {target_number} in {attempts} attempts.")
```

Part 07

Build a Pong-style game with Raspberry Pi

Create a version of the classic pong game using a Raspberry Pi. This project creates rotating paddles using potentiometers and an SPI analogue-to-digital converter

MAKER

Stewart Watkiss

Also known as Penguin Tutor. Maker and YouTuber that loves all things Raspberry Pi and Pico. Author of *Learn Electronics with Raspberry Pi*.

penguintutor.com

twitter.com/stewartwatkiss

This game of bat and ball is based around a Raspberry Pi with potentiometers for the paddle controllers, to mimic the classic game from the 1970s. As a Raspberry Pi does not include any analogue inputs, an analogue-to-digital converter is used. This uses the Serial Peripheral Interface (SPI) which provides a way for a computer or microcontroller to communicate with integrated circuits and other devices. Python and Pygame Zero are used to create a graphical display with the Pong-style paddles and ball.

You'll Need

- MCP3008 ADC **magpi.cc/mcp3008**
- 2 × 10K potentiometers **magpi.cc/10kpot**
- Mini breadboards **magpi.cc/minibreadboardwhite**

01 Raspberry Pi

This project features a Raspberry Pi computer, which has an operating system and more processing power than Raspberry Pi Pico. It can connect direct to a screen using HDMI allowing us to create a graphical game. One thing it doesn't have are analogue inputs, which are included in a Raspberry Pi Pico.

This tutorial will show how an external analogue-to-digital converter (ADC) can be added to Raspberry Pi and used to control a game that makes use of Pygame Zero.

02 MCP3008 ADC

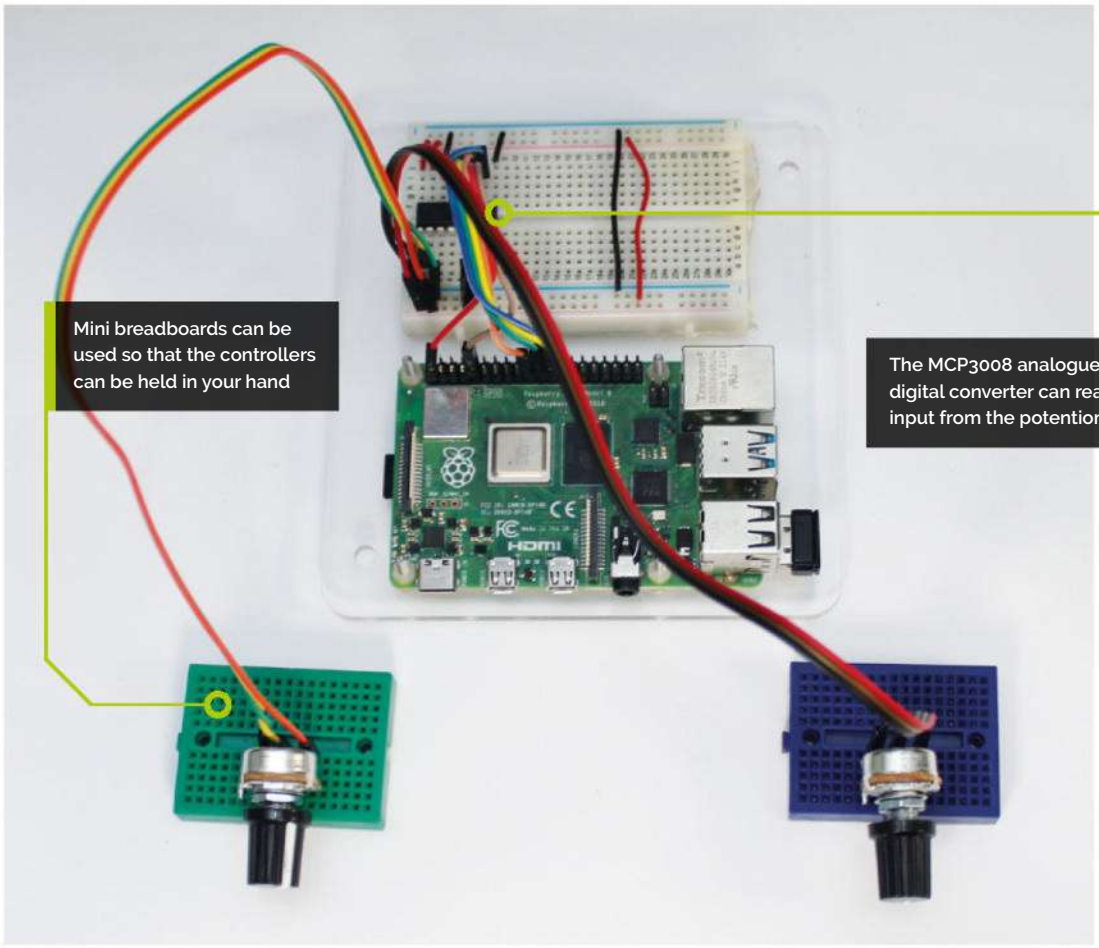
The MCP3008 is an integrated circuit which can sample analogue inputs and provide the values as a digital value for a Raspberry Pi or microcontroller. The MCP3008 has eight inputs which can be used individually, or as pairs, to provide voltage comparison. It converts the analogue value into an 8-bit number between 0 and 1023.

There are Python drivers available, but they are no longer supported. Instead, a simplified version of the library is provided in the GitHub repository: **magpi.cc/rpiponggit**. Save the file **mcp3008.py** into the same directory as your own code.

03 SPI

Raspberry Pi uses Serial Peripheral Interface (SPI) to communicate with the MCP3008. This is a protocol designed primarily for communicating with embedded systems. It is similar to I2C, but has an additional data line and requires a chip select connection for each peripheral device, instead of using device address numbers.

Figure 2 (overleaf) shows the connections between the controller and peripheral devices. The two data lines are labelled as SDO (serial data out) and SDI (serial data in). You may also see these referred to as MOSI and MISO. The CS lines are for chip select. These determine which peripheral the controller is communicating with. The line above CS indicates that the signal is inverted (it is active when set to a low signal).


Mini breadboards can be used so that the controllers can be held in your hand
The MCP3008 analogue-to-digital converter can read the input from the potentiometers

04 Wiring up the MCP3008

The circuit diagram is shown in **Figure 1**. The data out on your Raspberry Pi is pin 19 (GPIO 10 / MOSI), which is connected to the MCP3008 pin 11 (digital in). Data in on your Raspberry Pi is pin 21 (GPIO 9 / MISO), which is connected to the MCP3008 pin 12 (digital out). The clock signal is on pin 23 of your Raspberry Pi (GPIO 11), which goes to the clock input on pin 13 of the MCP3008. Finally, chip enable 0 is used on your Raspberry Pi from pin 24 (GPIO 8) to pin 10 on the MCP3008 (chip select). See **Figure 3** for the MCP3008 pinout. You should also enable SPI using the Raspberry Pi Configuration tool (**Figure 4**).

05 Wiring up the potentiometers

The power supply for the potentiometers and the MCP3008 comes from Raspberry Pi's 3.3V power supply. For convenience, this is connected to the bottom and top power rails on the breadboard. Looking from the top of the potentiometer, the left pin connects to ground and the right pin to the 3.3V supply. The wipers then

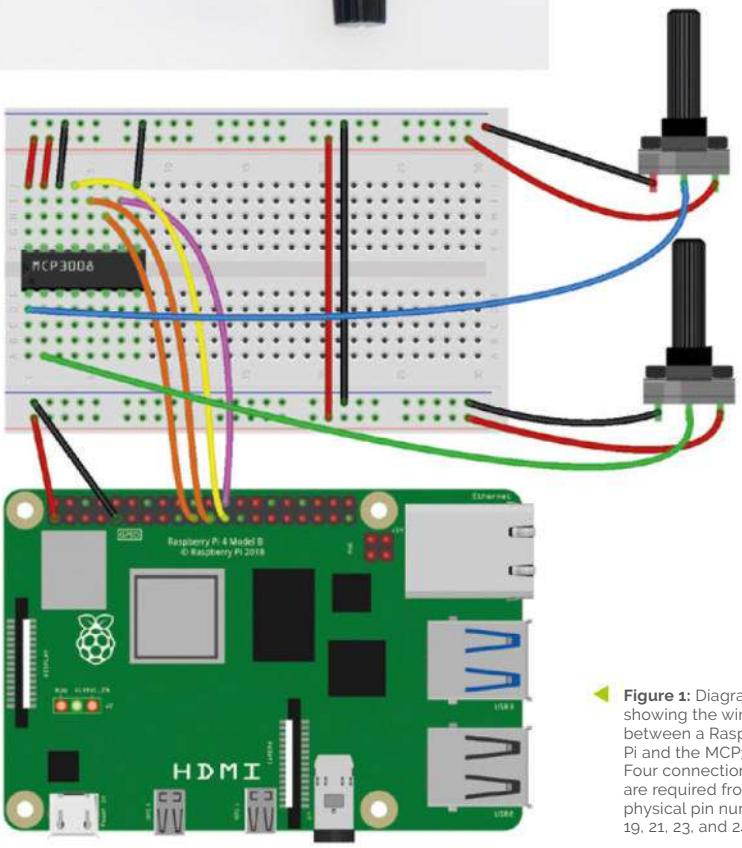


Figure 1: Diagram showing the wiring between a Raspberry Pi and the MCP3008. Four connections are required from physical pin numbers 19, 21, 23, and 24

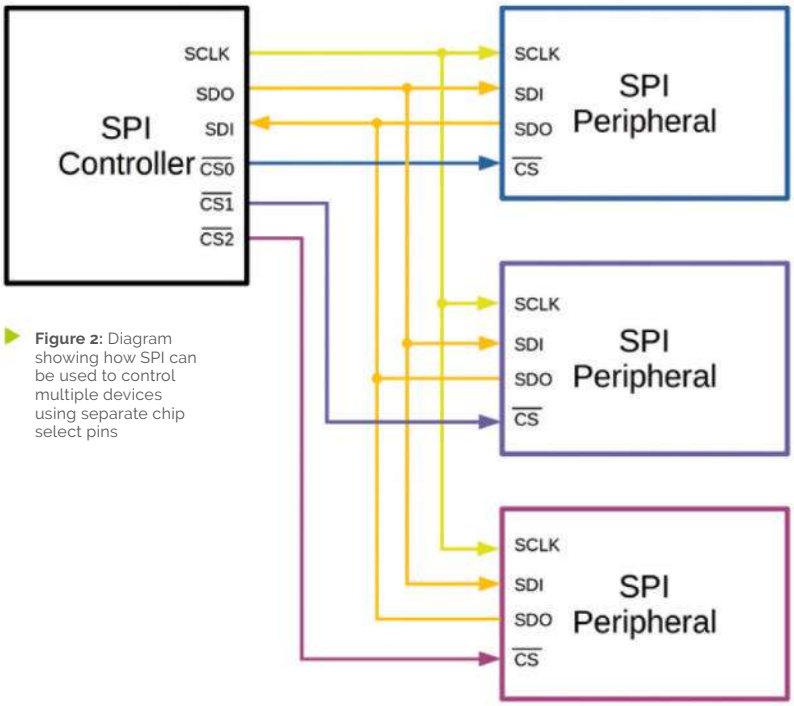


Figure 2: Diagram showing how SPI can be used to control multiple devices using separate chip select pins

Top Tip

SPI terminology

The terminology used is based on the latest guidelines from the Open Source Hardware Association. For more details see **magpi.cc/spisignalnames**.

connect to pins 1 and 2 of the MCP3008, which are the first two analogue inputs. This should provide 0V when the potentiometer is turned to the left. The wires could be soldered directly to the potentiometers, or jumper leads could be used to mini-breadboards.

06 Pygame Zero

The game needs a way to draw the graphics on the screen. One of the easiest ways of getting started creating graphics is with Pygame Zero. Using Pygame Zero, you can draw on the screen with minimal setup code.

Creating the game window is as simple as specifying the window dimensions using:

```
WIDTH = 1280
HEIGHT = 720
```

To run from the command-line, use `pgzrun` followed by the file name. You can also run the game using Thonny by selecting Pygame Zero mode under the Run menu.

07 Draw and update functions

To simplify code, Pygame Zero uses two functions, `draw` and `update`, each of which runs approximately 60 times per second. The `draw` function is used for drawing the screen. In this case: it clears the screen to remove the previous frame. It checks to see if the game has finished using `game_state` and if so displays a message to the screen using `screen.draw.text`. Otherwise, the game is in progress and it calls the functions to draw the two paddles and the ball.

The `update` method handles all the rest of the logic, including getting the position for the paddles from the ADC, updating the position of the ball, and detecting whether the ball hits the paddles or goes off the screen.

08 Get the paddle position

The position of the paddles is based on the reading from the MCP3008. This is read using:

```
adc0 = mcp.read_adc(0)
```

The value returned is between 0 and 1023, which needs to be converted to a value representing the position on a 720-pixel-high screen. This is done by dividing the value by a height constant of 1.5 and subtracting it from the screen height (as the y-axis is numbered from the top downwards). The size of the paddle is also subtracted to allow part of the paddle to go off the end of the screen.

This is an approximation, but allows the paddle to cover the full height of the screen without completely disappearing.

Note that the code refers to player 0 and player 1, but when displayed to the user they are shown as player 1 and player 2 respectively. This is to provide consistency with the Python lists which count from 0.

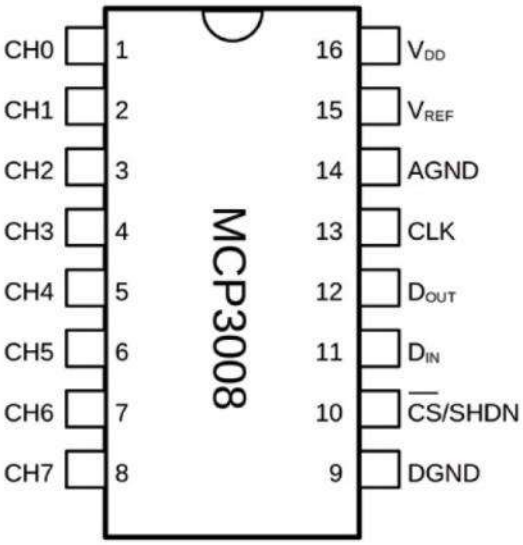


Figure 3: The pinout of the MCP3008 analogue-to-digital converter. The input channels can be used individually or in pairs

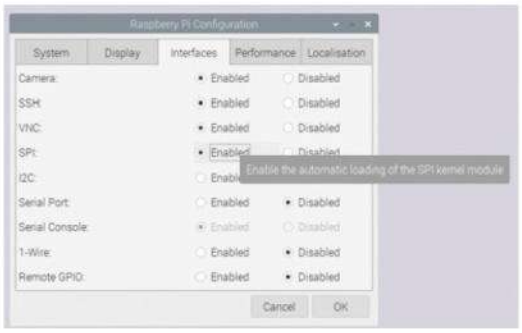


Figure 4: SPI needs to be enabled through Raspberry Pi OS's configuration tool under the Preferences menu

09 Creating the paddle rectangle

The paddle on the screen is drawn using a `Rect` object. This is a Pygame class which describes a rectangle. As well as being used for drawing the paddle, there are built-in methods that help with collision detection, which will be explained later.

The `Rect` object takes the x, y coordinates of the top left corner as the first argument and then width and height as a second. Player 1 has a `Rect` defined as below:

```
Rect((40, player1_pos), (paddle_width, paddle_height))
```

The value of 40 is the distance from the left of the screen and `player1_pos` is the vertical position. The width and height of the paddle are constant values but could be changed in a future version to change the difficulty. The `Rect` objects are stored in a global list called `paddle_rect`.

10 Drawing shapes in Pygame Zero

The paddles are drawn using the `draw_paddle` function which in turn calls the `filled_rect` draw method for the relevant player.

The ball is drawn using the `draw_ball` method, which draws the ball as a filled circle based on the ball's x and y coordinates.

In both these examples, the `color` variable is set to white to mimic the classic game, but they could be changed to provide different-coloured paddles and ball.

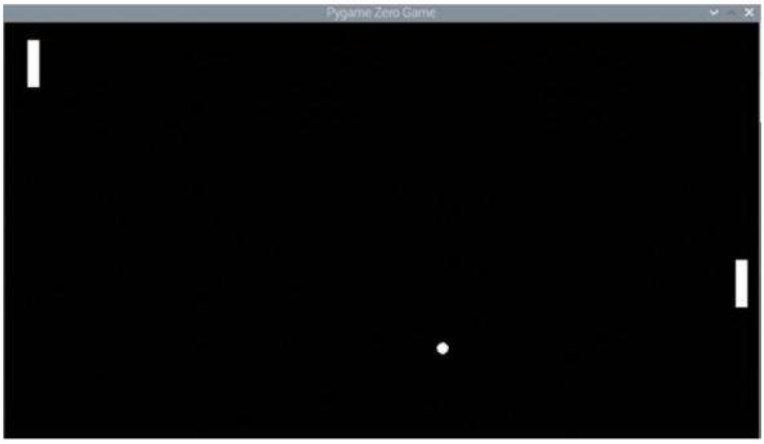
11 Game play

To turn this into a game, the ball needs to move around and detect when it's hit something. This is handled within the `update` function.

The ball has a list for the velocity which determines the speed of the ball in the x and y directions; this is added to the position of the ball.

The code then checks to see if the ball has hit the left or right side, in which case it sets the state to `gameover` and sets the winner. It then checks if the ball has hit a paddle, which is done using the `Rect collidepoint` method. If the ball has hit a paddle, the `deflect_ball` function is used to change the x direction and, depending upon where on the paddle, the speed in the y direction. It also checks to see if the ball has hit the top or bottom, in which case the y direction is changed.

Finally, the speed is increased each turn to increase the difficulty as the game is played.



The basic game with white paddles and a ball

A second version of the game is included in the GitHub repository showing the score and coloured paddles

12 Future improvements

At the moment, the game ends after a single point is scored. To improve the game, this can be expanded to handle scores and the score displayed on the screen. This is included in an updated version called **pong2.py** included in the GitHub repository.

You may wish to add the ability to run the game a second time, or even add different games. This may need a button for each controller so that players can select when to start the game. You could even create other games using the paddle controllers, such as a Breakout clone.

Top Tip

Pygame Zero

Pygame Zero is designed for education and makes creating your own games as simple as possible. For more information see **magpi.cc/pygamezero**.

pong.py

> Language: **Python Pygame Zero**

```
# Pong using MCP3008 and two potentiometers
# Run through Pygame Zero (pgzrun)
import time
from mcp3008 import MCP3008

WIDTH = 1280
HEIGHT = 720

mcp = MCP3008()

WHITE = (255,255,255)

player0_pos = mcp.read_adc(0)
player1_pos = mcp.read_adc(1)

#starting positions
ball_x = 640
ball_y = 360
ball_speed = 5 # default 5
ball_velocity = [1 * ball_speed, 0.5 * ball_speed]
ball_radius = 10

paddle_height = 80
paddle_width = 20

# Rectangle representing paddles
paddle_rect = [(0,0),(0,0)]

# Ratio screen (720) to adc (1024)
height_constant = 1.5

game_state = "play"
game_winner = 0

def draw():
    screen.clear()
    if (game_state == "gameover"):
        # To make more user friendly change player 0 to 1, and 1 to 2
        screen.draw.text("Player {} wins!".format(game_winner+1), (100, 50), fontsize=60)
    else:
        draw_paddle(screen, 0)
        draw_paddle(screen, 1)
        draw_ball(screen, ball_x, ball_y, ball_radius)

def update():
    global paddle_rect, player0_pos, player1_pos, ball_x, ball_y, ball_velocity, game_state, game_winner
    adc0 = mcp.read_adc(0)
    adc1 = mcp.read_adc(1)
    player0_pos = HEIGHT - int(adc0 / height_constant) - paddle_height
    player1_pos = HEIGHT - int(adc1 / height_constant) - paddle_height
    paddle_rect[0] = Rect((40, player0_pos), (paddle_width, paddle_height))
    paddle_rect[1] = Rect((WIDTH-40, player1_pos), (paddle_width, paddle_height))
    # Update ball position
    ball_x += int(ball_velocity[0])
    ball_y += int(ball_velocity[1])

    # Hits wall = game over
    if (ball_x - ball_radius <= 0):
        # player 1 wins if passes left
        game_winner = 1
        game_state = "gameover"
    elif (ball_x + ball_radius >= WIDTH):
        # player 0 wins if passes right
        game_winner = 0
        game_state = "gameover"
        # Collides with paddle - checks centre of the ball
    elif (paddle_rect[0].collidepoint (ball_x, ball_y)):
        deflect_ball (0)
    elif (paddle_rect[1].collidepoint (ball_x, ball_y)):
        deflect_ball (1)

    # Hit top or bottom (bounce)
    if (ball_y + ball_radius >= HEIGHT or ball_y - ball_radius <= 0):
        ball_velocity[1] = ball_velocity[1] * -1

    # Increase speed of ball - quick increase
    # Max speed 20
    if ball_velocity[0] < 20 and ball_velocity[0] > -20:
        ball_velocity[0] = ball_velocity[0] * 1.001
        ball_velocity[1] = ball_velocity[1] * 1.001

# player left = 0, right = 1
def draw_paddle(screen, player, color=WHITE):
    if player == 0:
```

DOWNLOAD THE FULL CODE:

magpi.cc/rpiponggit

```
085.          screen.draw.filled_rect(paddle_rect[0],
      color)
086.      else:
087.          screen.draw.filled_rect(paddle_rect[1],
      color)
088.
089.  def draw_ball(screen, ball_x, ball_y,
      ball_radius=10, ball_color=WHITE):
090.      screen.draw.filled_circle ((ball_x,ball_y),
      ball_radius, ball_color)
091.
092.  def deflect_ball (player):
093.      global ball_velocity, ball_x, ball_y
094.      # reverse horizontal direction
095.      ball_velocity[0] = ball_velocity[0] * -1
096.      # also move the ball 2 pixels away to
      prevent it getting stuck
097.      if player == 0 :
098.         ball_x += 2
099.      else :
100.          ball_x -= 2
101.      # if moving down
102.      if ball_velocity[1] > 0:
103.          if ball_y < paddle_rect[player].y +
      (paddle_height / 3):
104.              ball_velocity[1] = ball_velocity[1]
      * 0.5
105.          elif ball_y > paddle_rect[player].y + 2
      * (paddle_height / 3):
106.              ball_velocity[1] = ball_velocity[1]
      * 1.5
107.          # If central part of paddle then no
      change
108.      else:
109.          if ball_y < paddle_rect[player].y +
      (paddle_height / 3):
110.              ball_velocity[1] = ball_velocity[1]
      * 1.5
111.          elif ball_y > paddle_rect[player].y + 2
      * (paddle_height / 3):
112.              ball_velocity[1] = ball_velocity[1]
      * 0.5
113.          # If central part of paddle then no
      change
114.
115.
116.
117.
118.
119.
120.
```

Part 05

Camera Module: Controlling the camera

Take control of your Raspberry Pi Camera Module and how it captures images

David Plowman

David is an engineer at Raspberry Pi with a special interest in camera software and algorithms, and image processing hardware.

raspberrypi.com

Our previous Camera Module tutorials have spent some time explaining how to capture camera images in different ways, but we haven't yet shown how you can take control of the camera itself, and the images that it delivers.

There are a number of different ways that we can control the behaviour of the camera, so we're going to divide them into three groups:

1. **Image readout controls.** The orientation in which images are read out and how we can zoom into different parts of the image.
2. **Camera controls.** Various controls for the camera itself, such as the frame rate, exposure, white balance, and autofocus.
3. **Image processing controls.** Finally, we have a number of options for how the captured images can be tweaked, such as increasing the sharpness, contrast, or colour saturation of the image.

Normally the camera system gives you images that are 'the right way up'. But there are occasions where a camera may be mounted differently, or perhaps there is a reason why an application may want a 'mirrored' image.

Changing the image orientation

To mirror an image horizontally, use the option `--hflip`. This option applies identically to all our libcamera applications, for example:

```
libcamera-still --hflip -o mirrored.jpg
```

To flip an image vertically, use the `--vflip` option. We use it in exactly the same way as `--hflip`. Finally, to rotate an image through 180 degrees you can specify both `--hflip` and `--vflip`, or you can use the option `--rot 180` which has the same effect; for example:

```
libcamera-vid --rot 180 -o rotated.h264
```

Note that the libcamera applications are not able to rotate an image through 90 or 270 degrees. The easiest way to apply such a rotation to a captured image or video is to edit the file's metadata using a tool like ExifTool. For example, to rotate a JPEG image by 90 degrees, use:

```
exiftool -orientation="rotate 90" image.jpg
```

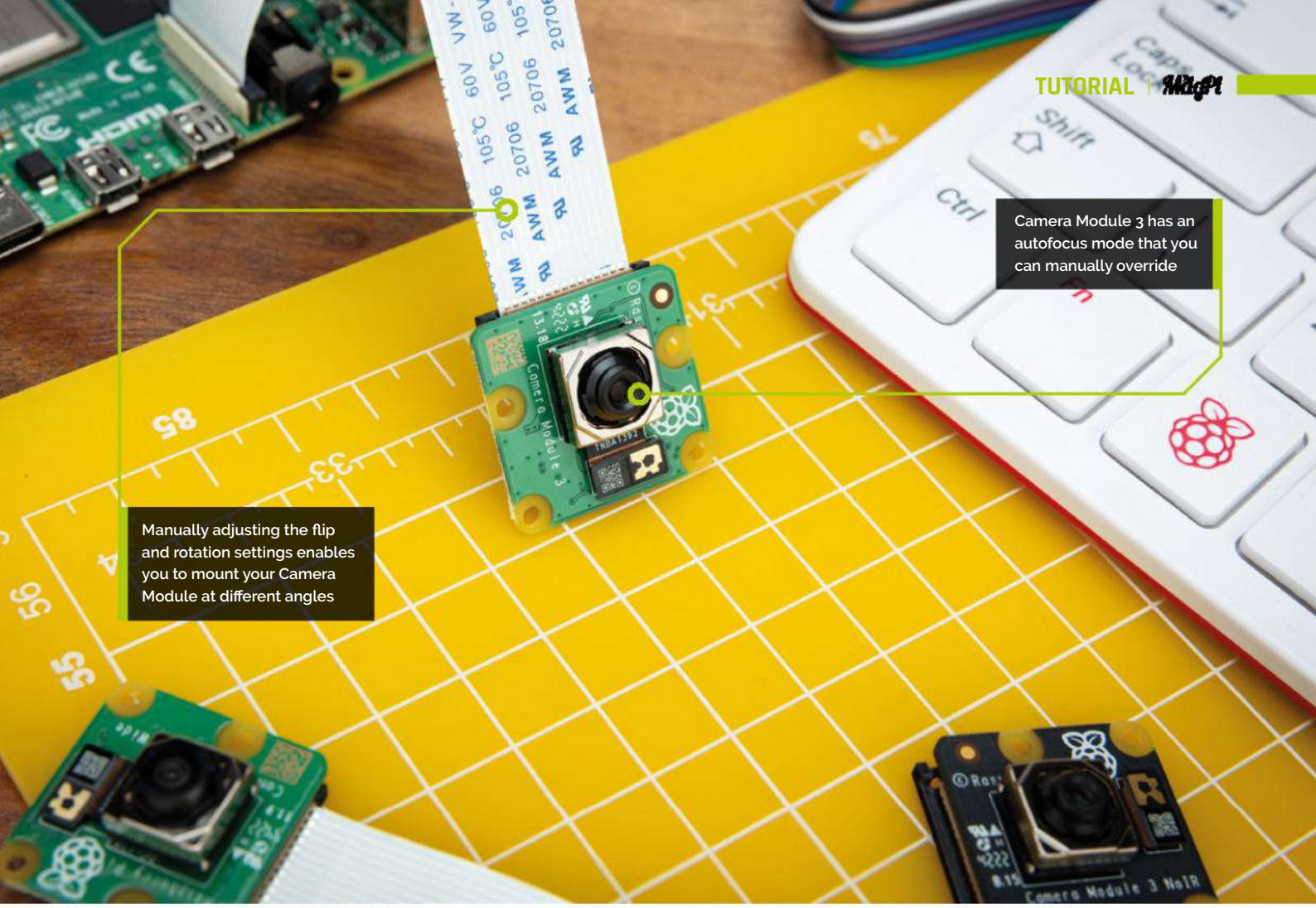
Or use `"rotate 270"` to rotate the image anti-clockwise.

To rotate an MP4 video through 90 degrees, use:

```
exiftool -rotation=90 video.mp4
```

Again, `-rotation=270` will rotate the video the other way.

However, it is not possible to rotate an H264 file in this way because, as we discovered in Part 3 ('More on video capture', **magpi.cc/131**), it is not a true container format.


Camera Module 3 has an autofocus mode that you can manually override
Manually adjusting the flip and rotation settings enables you to mount your Camera Module at different angles

> The units are a proportion of the full camera image

Digital zoom

Most cameras have a 'zoom' feature. There are two types of zoom: 'optical zoom', which requires a zoom lens, and 'digital zoom', which works without a zoom lens by using digital upscaling (though often this can result in reduced quality).

Raspberry Pi cameras implement digital zoom by processing only a rectangular subset (a 'crop') of the pixels that we receive from the image sensor. This crop is scaled back up to the full output size. By changing this crop, we can zoom into the image – and even, by sliding it around, pan across the image.

Our libcamera applications all allow you to specify a 'region of interest' parameter with `--roi`. This should be followed by a comma-separated list of four numbers:

- **x-offset:** The offset in the x-direction of the left-hand edge of the crop from the left-hand edge of the camera image.
- **y-offset:** The offset in the y-direction of the top edge of the crop from the top edge of the camera image.
- **width:** The width of the crop.
- **height:** The height of the crop.

In all cases, the units are a proportion of the full camera image. So, if you wanted to zoom into the centre of the camera image with a scale factor of 2, you would use:

```
--roi 0.25,0.25,0.5,0.5
```

Notice how the values 0.25 cause the crop to be centred in the original image; if we had used zero instead in both places we would be zooming into the top left corner of the original.

Let's finish with an example:

```
libcamera-hello --roi 0.25,0.5,0.333,0.333
```

This will zoom in by a factor of 3 onto an area where the top left pixel is a quarter of the way across the full camera image, and halfway down.

A useful tip is that to preserve the correct aspect ratio of the image, the width and height values should be the same (because they are relative to the original image from the sensor).

▲ The same image captured with `libcamera-still` and with the `libcamera-still --awb tungsten` option

Camera modes

Some image sensors may output images with a single resolution, and always showing the same field of view. Raspberry Pi cameras, however, generally give you a choice of resolutions. Sometimes these may all show the same field of view, or some of them may show a reduced field of view. Each of these is known as a 'camera mode'. How do you know what camera mode you should be using? Raspberry Pi's libcamera applications will all choose a camera mode for you based on the output image sizes that you request. However, the camera modes all have different trade-offs, so it can't guarantee to get this right all the time.

Camera modes, therefore, are a somewhat tricky topic, and will be the subject of the tutorial.

> "Many consumer digital cameras have familiar controls for things like the exposure time"

Camera controls

Many consumer digital cameras have familiar controls for things like the exposure time, the ISO (or gain, in a digital camera), the white balance and so on. All these are available for Raspberry Pi cameras too.

Normally the camera runs an automatic exposure algorithm, adjusting the exposure time in order to produce images of 'reasonable' brightness. But sometimes users may want images that are brighter or darker than the 'default'.

One convenient way to do this is with the `--ev` parameter. This parameter adjusts the target for the auto exposure algorithm in units of stops. The default value is 0, and adding 1 to the value doubles the brightness. Similarly, subtracting 1 halves the brightness, so it is a logarithmic scale. For example:

```
libcamera-hello --ev 0.5
```

…will increase the overall exposure by 0.5 stops, equivalent to $\sqrt{2}$, an increase of approximately 41%.

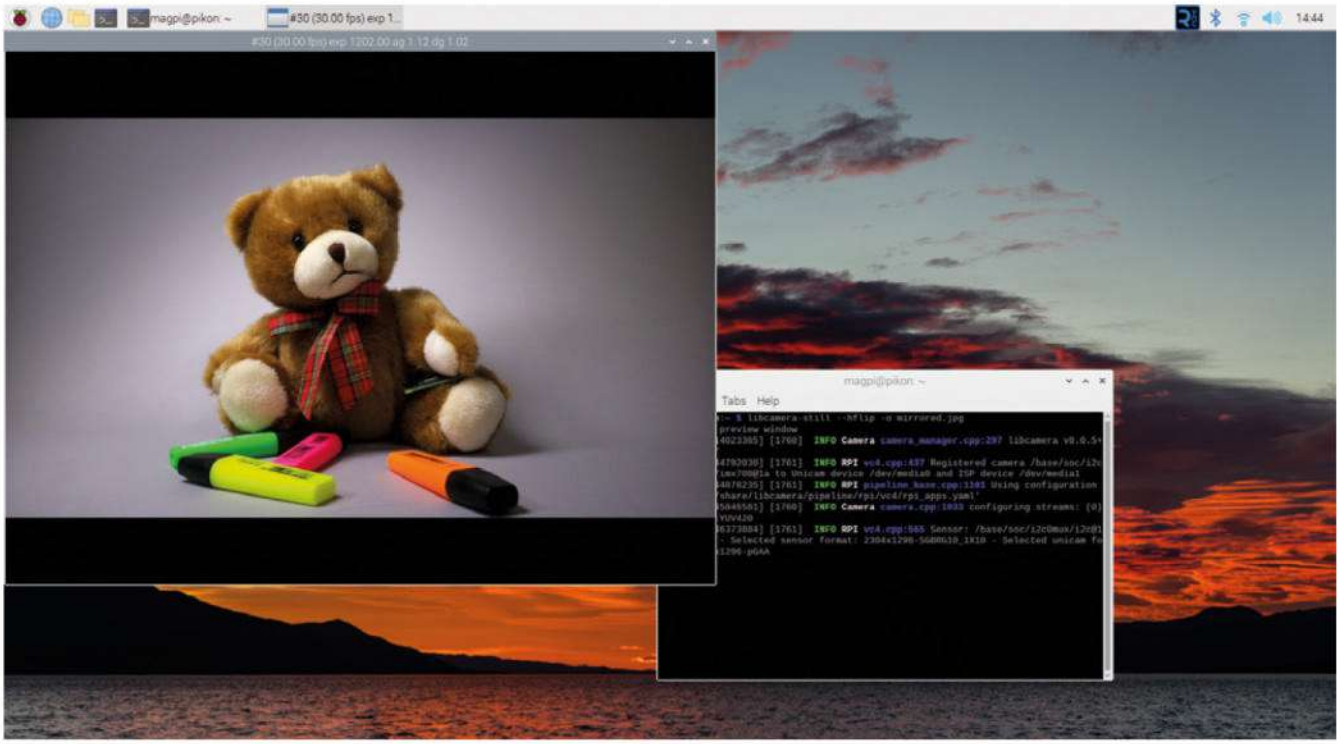
Exposure mode

Normally, the camera system delivers the target overall exposure with a balanced choice of values for the camera's shutter speed and the camera's gain. In some situations, such as when capturing moving objects, it can be desirable to lower the time for which the shutter is open, but to increase the gain to compensate.

This can be accomplished by requesting the 'sport' exposure mode:

```
libcamera-still --exposure sport -o action_
shot.jpg
```

In the end, you always have the option to select your own values for the camera's shutter speed and gain. If you set just one of them, then the other will still be varied by the auto exposure algorithm so as to give you a 'reasonable' exposure. But if you fix both, then the brightness of your images is completely determined and will never change.



A mirrored image of the scene captured using `libcamera-still --hflip -o mirrored.jpg`

The command line parameters that we use are `--shutter` to set the time for which the image is being exposed (in units of microseconds), and `--gain` to set the gain value. Multiplying the two together gives us the 'total effective exposure', so that any images with the same total effective exposure should be equivalently bright. Let's see an example:

```
libcamera-still --shutter 9000 --gain 2.0 -o
image.jpg
```

This will capture an image that has been exposed for 9ms, and with an analogue gain of 2.0. There will be no variation of brightness, whatever the camera is pointed at.

Frame rate

The cameras normally run at 30 frames per second (fps), but you can set a different value using the `--framerate` parameter, also in units of frames per second. For example, to run the camera at 15fps, you could use:

```
libcamera-hello -t 0 --framerate 15
```

You can use non-integer numbers of frames per second too. Note further that:

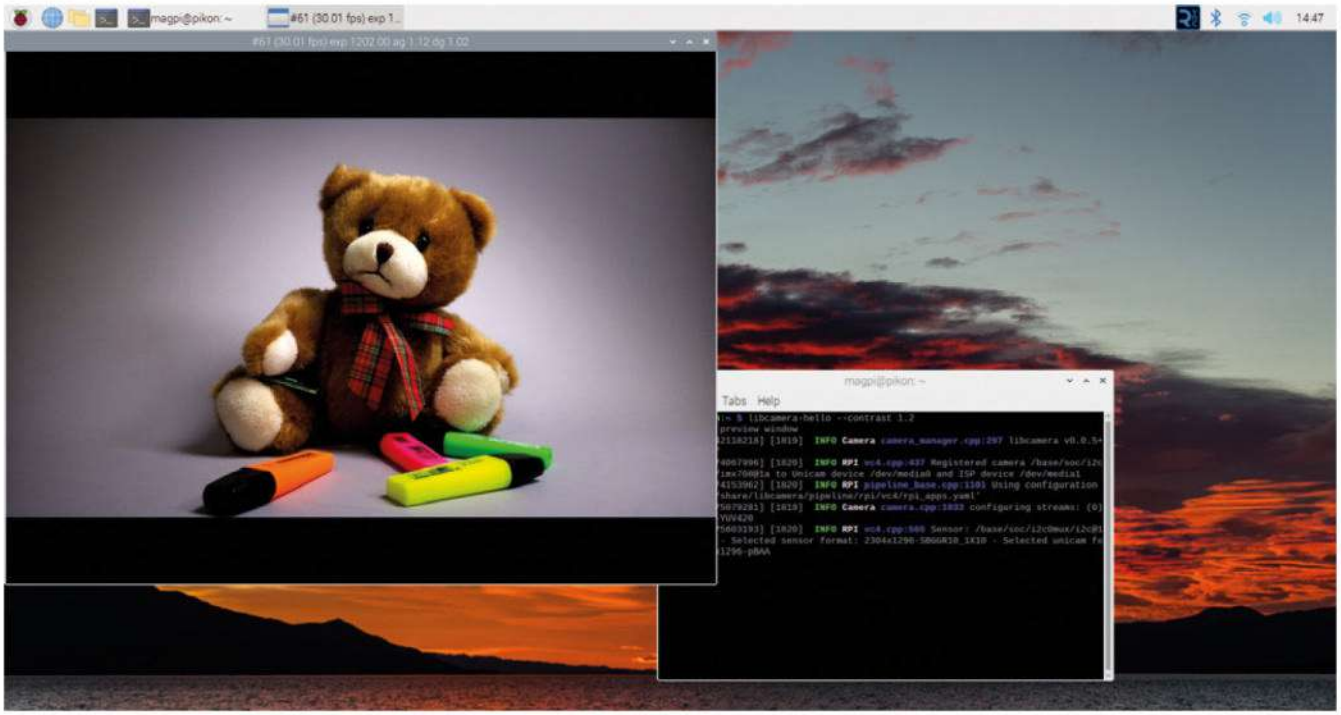
1. The frame rate implies an upper bound on the longest exposure that is possible. For instance, at a frame rate of 30fps, you won't be able to get exposures longer than 1/30s, or about 33ms. In practice, the limit will be very slightly lower than this for technical reasons.
2. Cameras will generally have a maximum and minimum frame rate that they support. If you ask for a faster or slower frame rate, this will be clamped to the maximum or minimum that is achievable. These limits normally depend on the camera and the 'camera mode', which will be the subject of a subsequent tutorial.

White balance

Setting the camera's white balance is another 'traditional' camera function, even though it is actually a function of the software that drives the camera, and not of the image sensor itself (as was the case with exposure, for example).

By default, the camera system uses the 'auto' white balance mode, which attempts to compensate for a very wide range of illuminants, ranging from very orange indoor lighting to very blue outdoor illumination. But you have a number of different modes that you can choose which work by constraining this range. We have:

- **auto:** Assume the illuminant may come from a very wide range of everyday sources (the default setting).
- **incandescent:** Assume the illuminant is some form of incandescent lighting.
- **tungsten:** Assume the illumination is provided by a tungsten filament.



An image captured using high contrast settings

- **fluorescent:** Assume the illuminant is some form of fluorescent light.
- **indoor:** Assume the camera is pointing at a scene with indoor lighting.
- **daylight:** Assume the scene is outdoor with daylight illumination.
- **cloudy:** Assume that the ambient conditions are a cloudy outdoor day.

You should set your choice with the `--awb` option. For example, to choose the daylight white-balance mode, use:

```
libcamera-still --awb daylight -o daylight.jpg
```

Colour gains

The AWB algorithm is really just choosing the red and blue gains that will be applied to the image to correct the colours. If you wish, you can set these values explicitly, though you will need some a prior knowledge as to what values will be appropriate.

For example, to set a red gain of 1.5 and a blue gain of 2.0, we can use:

```
libcamera-still -awbgains 1.5,2.0 -o image.jpg
```

The gain values are given in the order red followed by blue, and should be separated just by a comma (and no spaces).

In general, you can expect that the values you need to use will vary between different models of camera (for example, values that give a 'correct' image for a Camera Module v2 will give an image with a very pronounced colour cast with an HQ Camera).

Autofocus and lens position

Autofocus and a controllable lens are only available with the Camera Module 3. Other Raspberry Pi cameras, other than the v1 Camera Module, have lenses that can be controlled manually or (in the case of the v2 Camera Module) with a small tool that was supplied with the camera. The following section only applies to the Camera Module 3.

The lens position can be set explicitly using the `--lens-position` parameter. The units are 'dioptres', which is a reciprocal distance measure calculated by: lens position = 1 / focal distance in metres.

For example, to focus on an object two metres away, set the lens position to 1 / 2 = 0.5.

By convention, the value zero is taken to mean 'at infinity'. Note that the relationship between the lens position and the focal distance is only approximate. There is variation between different camera modules, variation with temperature, and it will also change significantly if the lens is pointed up or down.

For example, to capture a still image focused at ten metres away, use:

```
libcamera-still --lens-position 0.1 -o 10m.jpg
```

The lens is moved immediately to position 0.1, after which it will not move again.

The autofocus algorithm has three distinct modes:

- **manual:** The autofocus algorithm is disabled and the lens position must be set manually using the `--lens-position` parameter, after which it will not move again.
- **auto:** An autofocus sweep is performed when the camera starts, after which the lens does not move again. If the additional parameter `--autofocus-on-capture` (with no arguments) is added, then another sweep will be repeated just before any still image is captured.
- **continuous:** The lens will be moved continuously to find the best focus position at all times.

Normally the autofocus algorithm defaults to continuous mode, unless:

- The `--lens-position` parameter was specified, in which case it will be in manual mode.
- The `--autofocus-on-capture` parameter was given, in which case it will use the auto mode.

Let's try some examples. To capture a video with continuous autofocus:

```
libcamera-vid --autofocus-mode continuous -o autofocus.h264
```

And to capture a still image where we perform a second autofocus cycle just before the capture:

```
libcamera-still --autofocus-on-capture -o autofocus.jpg
```

> Brightening an image a lot can lead to them looking a little washed out

Brightness

The final category of controls are ones that affect how we render the image from the image sensor. For example, do we want to make it more colourful, sharper, or brighter?

Image brightness can be changed with the `--brightness` parameter. It accepts a numeric value between -1 and 1, where -1 means 'black virtually everywhere' and 1 means 'white virtually everywhere'. The default value, meaning 'normal brightness' is zero.

The brightness control works by adding a fixed positive (brighter) or negative (darker) value to all pixels.

Brightening an image a lot can lead to them looking a little washed out. Conversely, darkening them a lot can crush the dark areas of an image. Often the `--ev` parameter gives a better result than `--brightness`.

Example:

```
libcamera-hello --brightness 0.2
```

Contrast

Image contrast is changed with the `--contrast` parameter. It accepts a numeric value from 0 upwards, where 0 means 'no contrast at all' (the image will be grey almost everywhere) and 1 is the default meaning 'normal contrast'. Larger values produce more contrast in direct proportion.

The contrast control works by applying a gain (which is in fact the contrast parameter) to the difference between every pixel and the value of a grey pixel.

For an image with 20% more contrast, you could use:

```
libcamera-hello --contrast 1.2
```

Colour saturation

Colour saturation can be changed with the `--saturation` parameter. It accepts a numeric value from 0 upwards, where 0 means 'no colour saturation at all' (the image will be greyscale) and 1 is the default meaning 'normal colour saturation'. Larger values produce more colour saturation in direct proportion.

If you wanted only 90% of the standard colour saturation, you could use:

```
libcamera-hello --saturation 0.9
```

Sharpness

Image sharpness can be changed with the `--sharpness` parameter. It accepts a numeric value from 0 upwards, where 0 means 'no sharpening at all' (resulting in a soft image) and 1 is the default meaning 'normal sharpening'. Larger values produce more sharpening in direct proportion. If you wanted only 50% more sharpening, you could use:

```
libcamera-hello --sharpness 1.5
```

reBartender V0.1

Get yourself a Raspberry Pi-powered drinks dispenser with this cool setup by Seeed Studio and their reTerminal

MAKER

Peter Pan

An application engineer at Seeed Studio, he's also a maker in his spare time, so he fits right in with the maker atmosphere of Seeed Studio.

seeedstudio.com

You'll Need

- reTerminal DM **magpi.cc/reterminaldm**
- 150 mm of 20 mm × 20 mm aluminium frame
- 4 × right-angle brackets to fit
- Nuts and bolts
- 3D-printed parts
- 4 × 12 V DC peristaltic pumps
- 12 V power supply
- Various jars, pipes, metal straws, and drinks to fit
- STL and code files: **magpi.cc/rebtgit**

During the celebration of the Dragon Boat Festival at Seeed Studio, a booth was set up to serve drinks with an automated cocktail machine and named reBartender V0.1.

The prefix 're-' is commonly used to mean 'again', and Seeed Studio created a product line called reThings to evoke 'redefine', and includes products like reTerminal (which we reviewed in *The MagPi* 122 (**magpi.cc/122**) and the latest reTerminal DM (Device Master). They're both powered by Raspberry Pi Compute Module 4 and have a lot of industrial interfaces – perfect for attaching to pumps, pipes, and more to help serve some cocktails. Let's get mixing.

01 Cut the aluminium

The aluminium frame needs to be cut into 2 × 15 mm lengths for feet, 2 × 25 mm for the legs, and a 70 mm piece for holding reTerminal DM with DIN rail. The peristaltic pumps are just held on by cable ties.

See **Figure 1** for the basic build – the feet are laid lengthways on the desk, and right-angle brackets are used to attach the legs, which are orientated upright. The long piece bridges the two, again attached by the brackets, and that's where we'll attach reTerminal later. Be careful using power tools for this – not only are they dangerous, but cut metal can be very sharp.

Figure 1: The basic build of the frame, once done

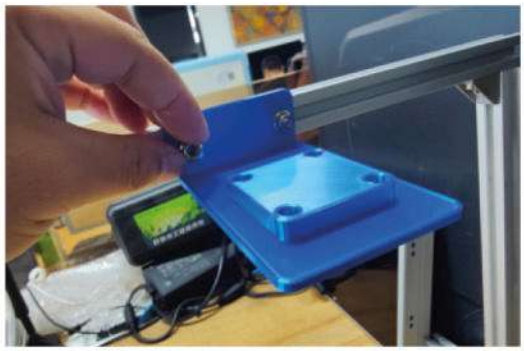


Figure 2: Affix the straw holder onto the aluminium frame

02 Attach pipe holder

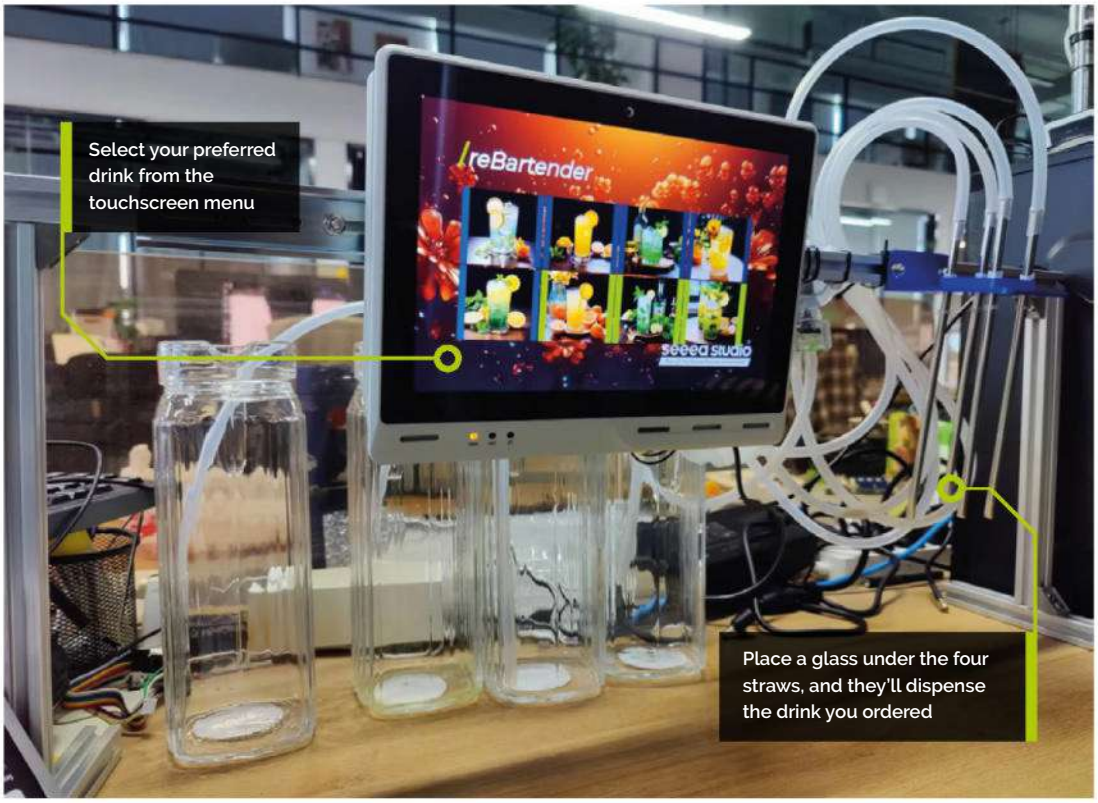
For the pipes attached to the pumps, the input end goes to a jug which will be filled with the various cocktail ingredients, and the output end of each silicon pipe goes into a metal straw. To hold the straws in a fixed position so that patrons can easily fill their glasses, a pipe holder is 3D-printed and attached to the frame.

Affix the short end to the frame with nuts and bolts (**Figure 2**), and then insert the straws into the pipes and through the holes on the 3D print.

> "Affix the short end to the frame with nuts and bolts"

03 Wiring it up

Figure 3 shows the wiring diagram for the setup. The DM Terminal is on the rear of reTerminal DM and has four opto-isolated DO (digital output) interfaces that can directly control a load of 500 mA. In this case, we can connected four peristaltic pumps with wiring to the reTerminal DM 20-pin terminal, where each individual pump can be controlled by the DO pins. The pumps used are driven by a 12 V DC motor, and the maximum current they draw is around 200mA.


Select your preferred drink from the touchscreen menu
reBartender
Place a glass under the four straws, and they'll dispense the drink you ordered

Warning!
Power tool safety

This project uses power tools. Be careful and wear safety equipment.

magpi.cc/powertools

04 Preparing to code

We're using Node-RED to create a 'proper cocktail mixing flow control algorithm', which is definitely real and important computer science, along with a nice UI for people to interact with.

The 20-pin terminal that we're using does correspond to specific GPIO pins on Raspberry Pi. You can find out more details on the entire terminal here: **magpi.cc/rtdmpins**. For our uses, the DO pins are labelled as such:

DO1 – GPIO21
DO2 – GPIO25
DO3 – GPIO26
DO4 – GPIO6

The fifth pin that we've plugged into is a ground pin for the DO pins.

05 Setting a pin in Node-RED

From the Raspberry Pi nodes, select rpi-gpio and add it to the flow. Double-click it to change the settings: pin needs to be 18 – GPIO24, BCM GPIO should be 24, type is Digital output, initialise pin state should be ticked, and initial level of pin should be low (0). You can double-check the settings next to **Figure 4** and, once you're happy, click Done. Repeat this for GPIO25, 26, and 06.

06 Flow control

Let's create the main algorithm. We can use the 'trigger node' to send messages with a short delay – the core algorithm behind the ingredient volume control. We can use 'change node' to send the stop signal after the delay period when chaining together more pumps.

In the settings for the trigger node, select 'true' for Send. Select 'wait for' for then and set the waiting period. This version uses 1000 milliseconds as it's about a 400 millisecond delay with the pump used to fill up a single shot glass. Set 'then send' to

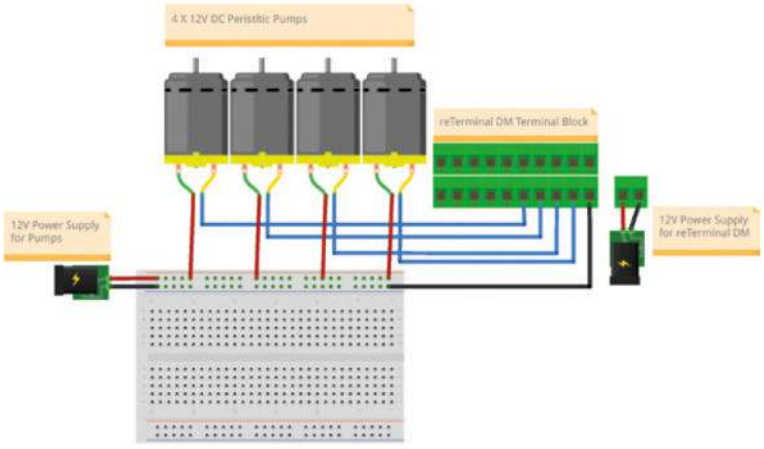


Figure 3: The wiring is fairly simple. Make sure the ground for the pump power is connected to reTerminal

Top Tip

Cocktail trial and error

Got a favourite cocktail? It usually has a very specific mixture of spirits and mixers, so getting the time delay correct for the right amounts will take some tweaking.

'true' for triggering next pump(/ingredient) within the same recipe. Tick 'send second message to separate output' and then hit Done.

For the change node, you'll need to open the properties and choose 'Set', 'msg.payload', and then to the value is 'false'. This allows one ingredient to be dispensed at a time.

07 Button nodes

Add a button node and open the properties. Set the group to '[HOME] Default', set the size to 6 × 5, and scroll down to 'When clicked, send' – set the payload to 1 which will be the number the button sends when pressed. Click Done.

We need to let Node-RED know we're using the UI to control stuff. Add a ui control node, open the settings and select JSON (or {}) as Send and then enter {"tab":"Overlay"}. Choose 'wait for' from the then menu, and enter 10000 milliseconds.

'Then send' should be JSON/{} again and this time enter {"tab":"HOME"}. You can check what you've done against **Figure 5**, and then click Done. Finally, add a ui control node, and set its output to 'Change tab or group events only'.

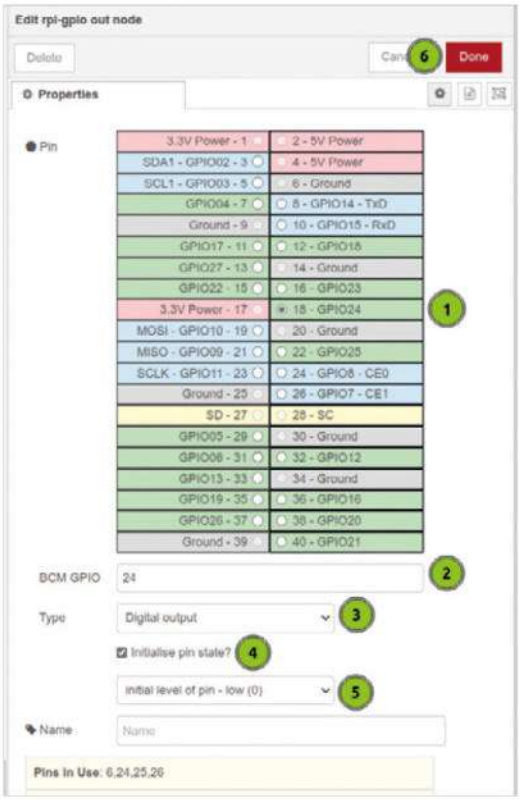


Figure 4: How to set up the GPIO pins in Node-RED

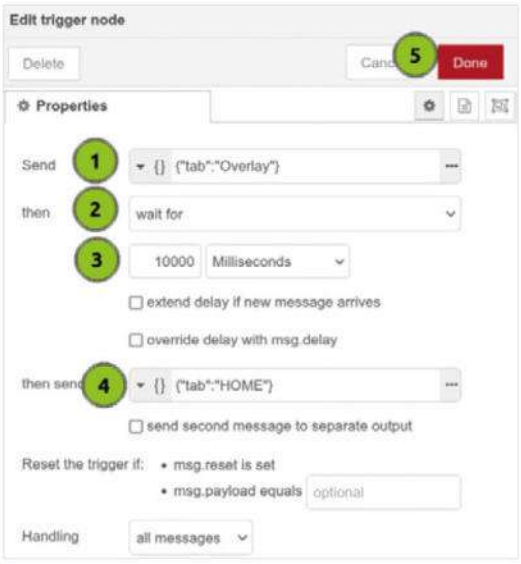


Figure 5: The settings for the trigger node

08 Record button presses

Add a csv node to the flow and open up the settings. For this, we want the Columns to be button, button_num, and comma as the Separator. In the input section, we want 'parse numerical values' ticked. Click Done when you're happy.

Now we need to write to the file. Add a write file node and set the file name to a path called test.csv. The Action should be append to file and then 'Add newline (\n) to each payload?' should be ticked. You want to now link the output of the csv node to the input of the write file node.

09 Main flow

With the nodes set up, we can start creating recipes by linking up the nodes! Check **Figure 6** for an example recipe, in this case a vodka and soda. With several recipes added to the flow, you'll get a flow like **Figure 7**. The recipes use sparkling water (pump one), vodka (pump two), orange juice (pump three), and mint syrup (pump four). Here are the original recipes:

Recipe 1 (Vodka and Soda): Sparkling water × 1000ms, vodka × 400ms
Recipe 2 (Screwdriver): Vodka × 400ms, orange juice × 800ms
Recipe 3 (Vodka Mojito): Sparkling water × 800ms, vodka × 200ms, mint syrup × 400ms
Recipe 4 (Orange Crush): Sparkling water × 800ms, vodka × 200ms, orange juice × 400ms
Recipe 5 (Vodka Collins): Vodka × 200ms, orange juice × 800ms, mint syrup × 400ms
Recipe 6: Sparkling water × 1000ms, orange juice × 400ms

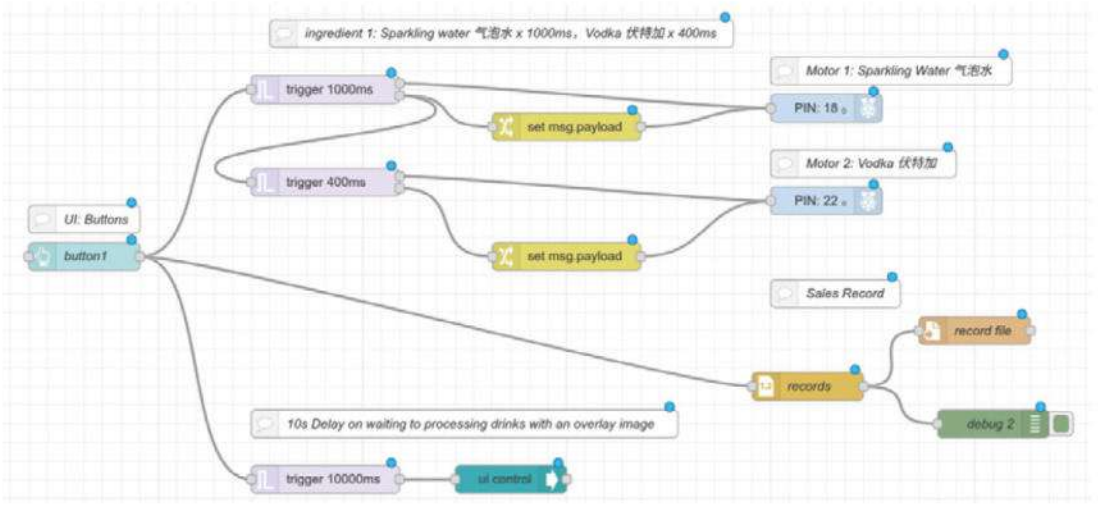


Figure 6: An example recipe flow

> We can set the buttons and other parts of the interface to use images

Recipe 7: Sparkling water × 1000ms, mint syrup × 400ms
Recipe 8: Sparkling water × 1000ms, orange juice 200ms, mint syrup × 200ms

10 UI folders

To make a nice-looking UI, we can set the buttons and other parts of the interface to use images. This is quite simple to do – first, open the terminal on Raspberry Pi and get to the Node-RED settings folder with:

```
cd ~/.node-red
```

Then open the **settings.js** file with:

```
nano settings.js
```

There will be the line `//httpStatic: '/home/pi/node-red-static/', //single static source`. Remove the // at the start to uncomment the line and save the file. Finally, use `cd ~` to head to the home folder and use:

```
mkdir node-red-static
```

Put the images for the drink buttons and background you'd like to use in the folder, and give them a simple name – e.g. button1, button2, background, etc.

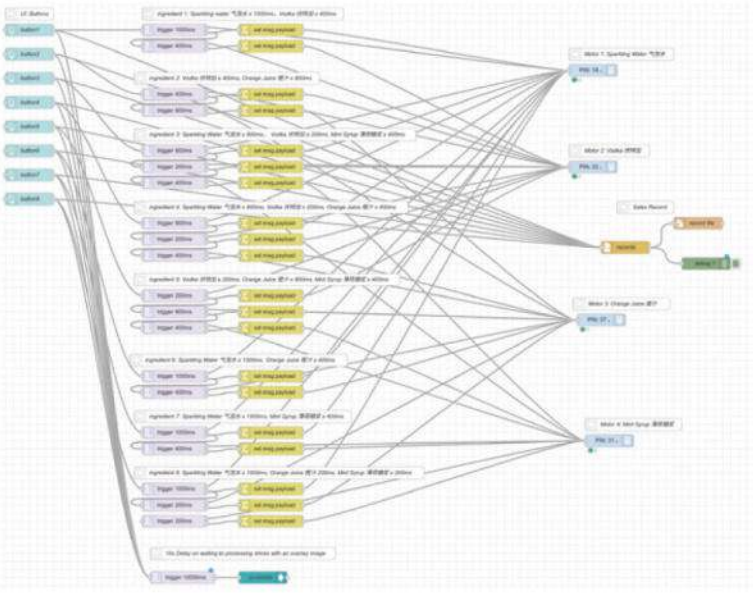
11 UI code

With that all done, you can add a template node to the flow and add CSS styling, linking the node id of the buttons to the images, and including the background. You can find the code used in this project at **magpi.cc/rebtgit**, as well as some example images if you're having trouble finding some yourself.

12 Automated sip

You're done! Fill up the jugs with the desired ingredients, and get ready for your automated bartender to serve you some drinks. Remember, drink responsibly – that way you can keep the containers filled up without any spills or accidents.

Figure 7: Full flow for eight buttons using four ingredients



Work & Learn with Raspberry Pi

Use your Raspberry Pi as a desktop PC for work and study

By Phil King

It may be small, but a Raspberry Pi single-board computer can be used as a desktop PC replacement for working from home. For this, we explore the best office applications and their online equivalents, plus how to email, instant message, and video chat with remote colleagues.

It's also important to make sure you're sitting correctly at your desk, so check out our ergonomic working tips. Finally, we take a look at learning to code and using Raspberry Pi to further your computer science skills.

Your Raspberry Pi office

Get working with LibreOffice and other productivity tools

Available in Raspberry Pi OS, the LibreOffice productivity suite features applications for word processing (Writer), spreadsheets (Calc), presentations (Impress), vector graphics editing (Draw), mathematical formulae (Math), and databases (Base).

If you have the Full version of Raspberry Pi OS, LibreOffice is among the recommended applications pre-installed and ready to use. From the desktop, click on the menu icon (raspberry) in the top-left of the screen and select Office and then the LibreOffice app you want to use.

If LibreOffice isn't already installed, you can use the Recommended Software tool (in Menu > Preferences) to install it. It will then appear in the Office submenu.

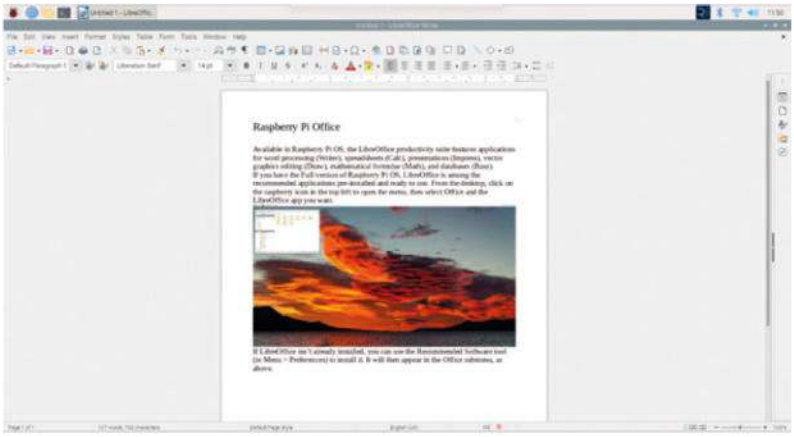
LibreOffice apps

LibreOffice features six applications for different work tasks. Let's take a look at them.

Writer

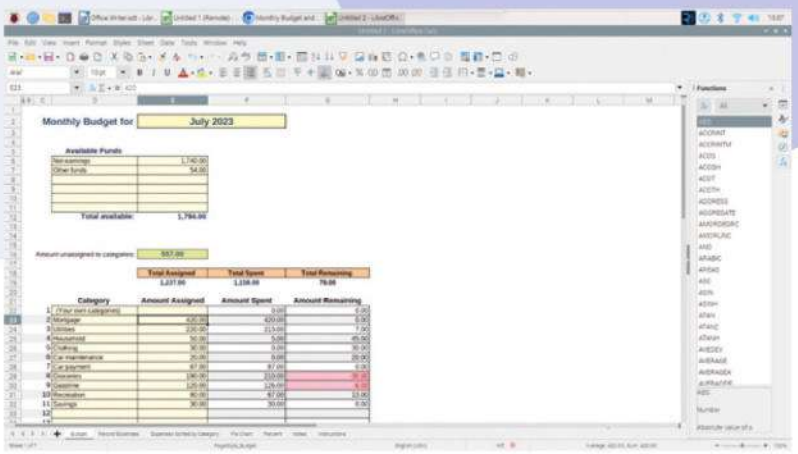
You'll feel at home with this word-processing app if you've ever used Microsoft Word or similar. Along the top of your text document is a standard toolbar with icons for commonly used functions; the formatting toolbar just below it includes options such as paragraph style, font, text size, colour, and alignment. There's also a pull-out sidebar on the right of the document for handy access to text properties, page format, paragraph styles, art gallery, and navigator. A status bar at the bottom shows info such as word/character count, page style, and language, along with a slider to zoom the view.

Want to insert an image into your document? From the top menu, select Insert > Image and browse to it in your Raspberry Pi's file system. You can also insert charts, media (including audio or video), objects (such as a formula), and shapes. Right-clicking an image brings up a menu of options including alignment and (text) wrap, or you can alter settings in the Properties panel in the sidebar.



You can add images to your Writer documents

If you don't want to start creating a document from scratch, there are a few templates to use; select File > New > Templates to view them. They can be filtered by type. For a more varied template selection (including for the other LibreOffice apps), visit the official Template Centre (**magpi.cc/libretemplates**) and download one, then open it in LibreOffice and save it as a template (File > Templates > Save as Template).



A household budget example in LibreOffice Calc

LibreOffice guides

For help getting started with LibreOffice and its apps, check out the official guidebooks, downloadable as free PDFs: **magpi.cc/librebooks**. For troubleshooting and tips, it's also worth visiting the LibreOffice online help page: **magpi.cc/librehelp**.

Calc

If you've used Microsoft Excel or similar, LibreOffice's spreadsheet app will feel familiar. It's ideal for tasks such as doing your business accounts, household budgeting, and organising tables of data for any purpose. It's also invaluable for opening CSV (comma-separated value) files used in coding projects.

As with Writer, along the top, you'll find standard and formatting toolbars, which can be customised if you need. There's also a formula bar for entering formulas using a wide range of functions – which can be viewed and filtered by type in the sidebar on the right.

The spreadsheet columns are labelled with letters along the top, while the rows are numbered on the left. So you can refer to each cell by its letter and number, e.g. C4 and F7, and use these in formulas for other cells. For instance, the formula **=SUM:(E23:E32)** adds the values in the range of cells E23 to E32.

Note that a spreadsheet may comprise more than one worksheet. Sheets are shown along the bottom and you can click the '+' to add and name a new one. You can also refer to cells on other sheets – and even documents – in formulas etc. Another useful tool is the 'pivot table' (Insert > Pivot Table) to combine, compare, and analyse selected parts of a large data set.

A chart wizard (Insert > Chart) helps you to create a chart based on the data in a selected range of cells. Chart types include column, bar, pie, area, and line – some are more suited to certain types of data. There are also lots of cell formatting options, including 'conditional' which changes according to a cell's value.

Impress

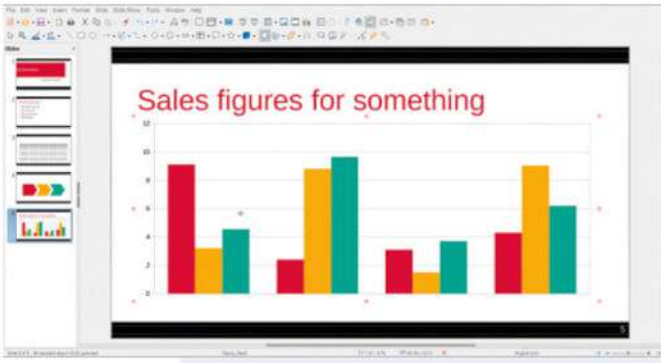
LibreOffice's presentation (slide show) app is the equivalent of Microsoft PowerPoint. By compiling a series of slides, you can create an effective and impressive work presentation – or maybe a slide show of your family photos.

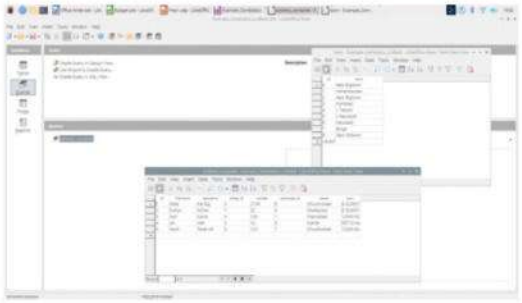
As well as including text elements such as bulleted lists and tables, you can make slides more visually appealing by adding photos, drawings, and even animations. Make your presentation look slicker, too, with a range of transition effects between slides, including swipe, fade, dissolve, and Venetian blind – available in the sidebar.

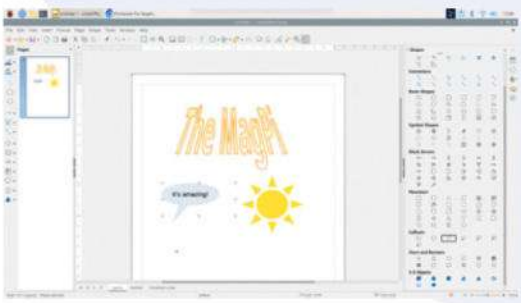
To make it easier to start creating a presentation, you're given the option of using any of several templates when you start a new document. A slide's basic layout can also easily be changed by selecting a new one in the sidebar Properties / Layouts panel.

Taking it further, you can create 'image maps' (Tools > ImageMap) that assign certain parts of an image (hotspots) to a web link (or additional text information). This is great for creating educational presentations.

Create colourful slide-show presentations with Impress







Create vector graphics and diagrams with LibreOffice Draw

LibreOffice Base is a fully-fledged database program

Base

Along with the familiar word processing, spreadsheet, and presentation office apps, LibreOffice features a fully-fledged database application, akin to Microsoft Access. As well as enabling you to organise your data in tables, it offers user-friendly forms for data entry. In addition, built-in wizards help you to get started creating a database, tables, and other elements.

Queries are a powerful feature, enabling you to filter database records and access the exact information you're looking for. Reports, meanwhile, are used to extract key data from the database and present it in an easy-to-read format for colleagues.

Another handy feature is the ability to filter and use names and addresses in a mail-merge operation to automatically create a mailing list for customers. Base can connect to other documents such as spreadsheets, text documents, and address books.

By default, Base uses an embedded Java-based HSQLDB relational database engine, but can also connect to many other common database types, such as MariaDB, MySQL, and PostgreSQL.

To help you get started with Base, read the official guidebook, downloadable as a free PDF (**magpi.cc/librebooks**). There are accompanying sample databases too – download the zip file from **magpi.cc/basesamples**.

Draw

While not as sophisticated as Inkscape, this vector graphics application enables you to draw lines, curves, shapes, arrows, and connectors, useful for creating diagrams and flowcharts. There's also a Fontworks tool with various styles for producing fancy text effects. It's also possible to extrude and edit 3D objects. Best of all, you can copy and paste your creations straight into other LibreOffice documents.

Math

This formula editor is a boon for maths and science students, enabling you to create complex formulas and equations which you can copy and paste into other LibreOffice documents. Terms can be entered in markup language, e.g. 'a over b', or you can right-click in the editor for a context menu, or select elements from a sidebar panel. It takes a little getting used to, but proves effective. Download the official Math guide book from **magpi.cc/librebooks**.

LibreOffice Math enables you to create sophisticated formulas

Online office

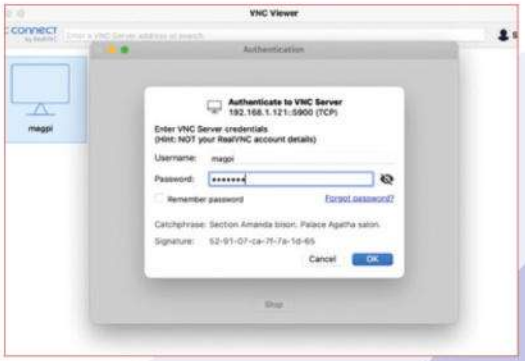
Whether you use LibreOffice or not, you may well want to share documents online and collaborate with colleagues. This is easily achieved via Raspberry Pi OS's Chromium web browser. Just sign into your Google Drive, Microsoft 365 or other cloud account, and you can access all your online documents and others shared with you. You'll also be able to upload LibreOffice documents and download others to edit in it – it works with most standard file formats.

Remote connection

Being able to access and control your Raspberry Pi from a remote computer (or smart device) can come in very handy. There are two main ways to do it.

SSH (Secure Shell) gives you a remote terminal to enter text commands. You can enable it in the Interfaces tab of Preferences > Raspberry Pi Configuration, or in the raspi-config tool from the command line. On another computer, open a Terminal and enter `ssh <user>@<IP>` – for example, `ssh pi@192.168.1.121`. Enter the password to connect.

VNC (Virtual Network Computing) gives you full access to Raspberry Pi's GUI desktop. Enable VNC in the Interfaces tab of Preferences > Raspberry Pi Configuration (or in raspi-config). On another computer, download the VNC Viewer application from **magpi.cc/vncviewer** and launch it, then log in with your Raspberry Pi's IP address, username, and password.



Communication with Raspberry Pi

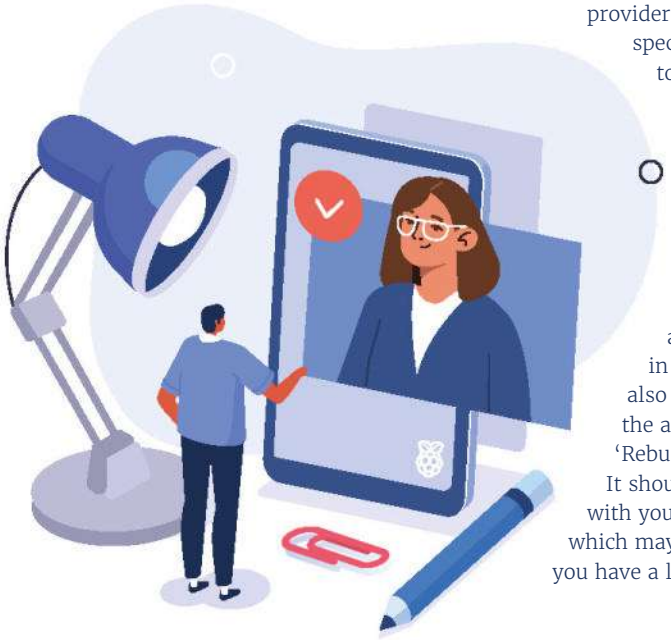
Stay in touch with colleagues and friends

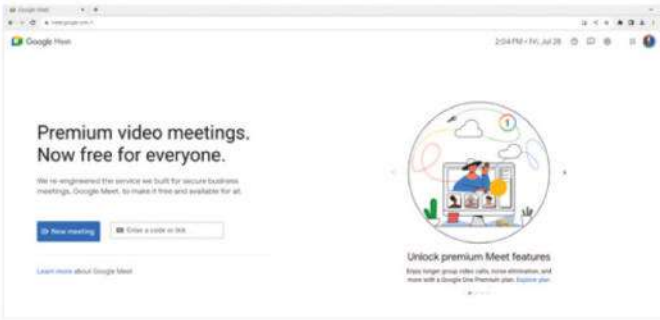
Naturally, you'll want to communicate with colleagues and associates while working or studying from home. Let's take a look at some of the options.

Email

Several email clients are available for Raspberry Pi. The default option is Claws, which comes pre-installed in Raspberry Pi OS – find it in Menu > Internet. (If Claws is not there, you can install it from Recommended Software). Other email clients include Thunderbird, Geary, and the terminal-based Mutt. Alternatively, you can log into a webmail service such as Google or Outlook via the Chromium browser.

Claws has a handy setup wizard to help you add an existing email account, with an 'autoconfigure' option. If you're using two-factor authentication, however, you may need to get your webmail provider to generate a special app password to use in place of your usual one. For instance, when using Claws with Gmail, check out the instructions at **magpi.cc/clawsgmail**. If you change the account settings in Claws, you may also need to right-click the account and choose 'Rebuild folder tree'. It should then populate with your email messages, which may take some time if you have a lot of them.





You'll need a camera and mic for video conferencing

Video conferencing

To use your Raspberry Pi for video conferencing, you'll need a Raspberry Pi Camera Module or USB webcam. If your camera lacks a microphone, you'll need to add that via a USB adapter/connection and assign it as an audio input. You can output audio via HDMI, or use a speaker or headset to hear what other people are saying. Audio input and output levels may be adjusted in Alsamixer – open a Terminal window and enter: `alsamixer -m`.

> **" You'll need to use the Chromium web browser to join the video chat "**

You'll need to use the Chromium web browser to join the video chat on whatever service you're using – such as Google Meet, Zoom, or Slack.

Instant messaging

The best way to use most instant most messaging services – such as Skype, Slack, or WhatsApp – is to access them via the Chromium web browser.

Sitting Comfortably?

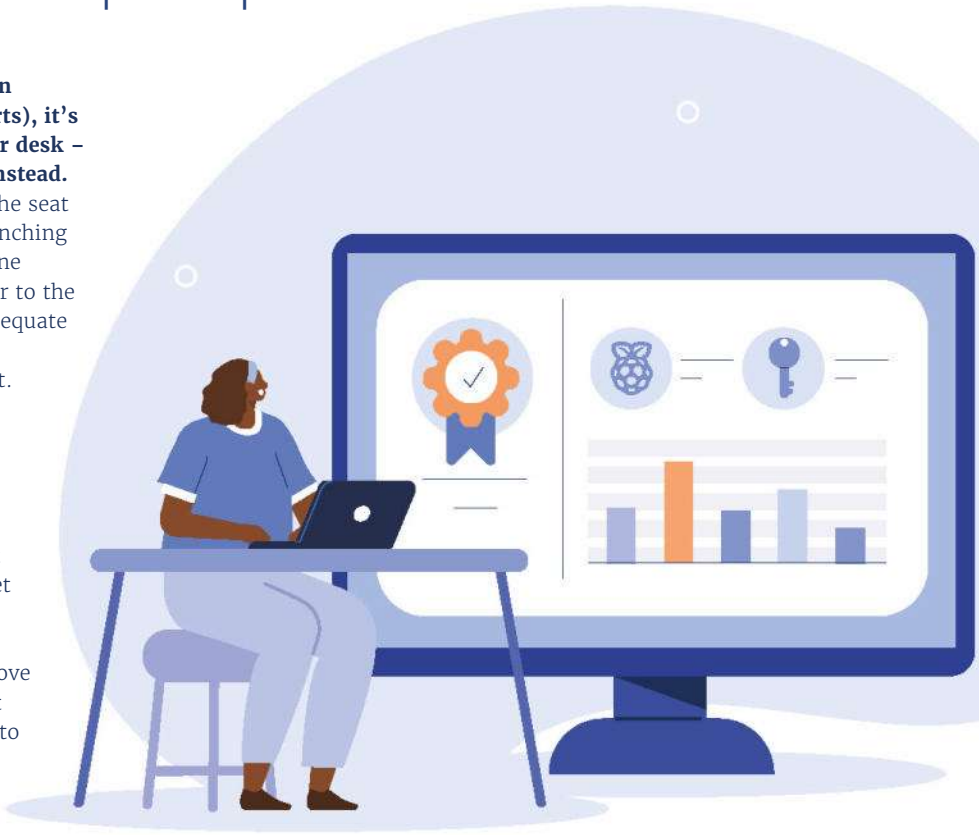
An ergonomic desk setup is important

Unless you want to end up with an aching back (and other body parts), it's important to sit correctly at your desk – unless you opt to use a standing desk instead.

The relative heights and positions of the seat and desk are both important to avoid hunching over your desk or leaning back. Your spine should be upright and thus perpendicular to the desktop. The seat should also provide adequate lower back and spinal support.

The monitor position is also important. Ideally, the top of the monitor should be around eye level so that you are looking down slightly at the centre of the screen. The monitor should generally be positioned at a distance around that of your fully extended arms. You might opt to use an ergonomic monitor stand to get the positioning just right.

Your elbows should be at an angle of around a 90° with your forearms just above desk height – rest your them on the seat arms if it has them. You shouldn't need to stretch to reach the keyboard, and you should aim to keep your hands in a neutral position: flat and pointing towards the monitor – the same principle applies to using a mouse. There's also the option of an ergonomic keyboard. As for wrist rests, the jury is still out about whether they are beneficial or not.



▼ A good office chair should support your spine and lower back

Legs should be kept uncrossed with feet flat against the floor – if you have trouble reaching it, you could add a foot-rest.

Finally, make sure you take regular work breaks and go get a little exercise!

Learn with Raspberry Pi

Your Raspberry Pi is an excellent learning tool

Being so low-cost, Raspberry Pi is ideal for experimentation and learning, without the worry of messing up an expensive desktop or laptop PC.

In addition to being a great general-purpose Linux-based computer for learning programming, Raspberry Pi has a superpower: its GPIO header. This enables it to be connected to custom-built electrical circuits and add-on boards, making it ideal for a huge variety of DIY electronics projects – as our regular Project Showcases section demonstrates.

With its wireless connectivity, Raspberry Pi can also run a web server which can be used to self-host all sorts of services for file/photo storage, media streaming, and blogs.

You can even use the CamJam EduKit #3's box as a chassis for your robot

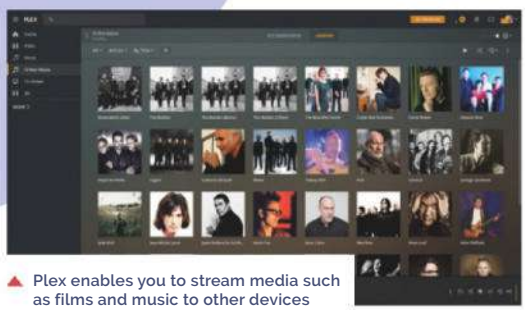
Project Idea: **Robot**

Raspberry Pi's small size makes it ideal for use in robotics projects. You can mount one on a wheeled chassis and then remote-control it wirelessly from another computer or a phone.

So far so RC car, but it gets more interesting when you program your Raspberry Pi robot with AI to make it self-driving, using proximity sensors to avoid obstacles. You can also add one or more downward-facing IR sensors so it can sense a dark path (made with tape or a maker pen) beneath it and follow it automatically – it's magical seeing your robot in action, thinking for itself.

A great way to get started with robotics is with the low-cost CamJam EduKit #3 kit (**magpi.cc/edukit3**). It's featured in our Simple Robotics feature and tutorial starting in issue 120 (**magpi.cc/120**).

Alternatively, you could get a grasp of robotics by building your own robot arm with a kit such as the MeArm (**magpi.cc/mearmmaker**). Check out our step-by-step tutorial in issue 131 (**magpi.cc/131**).



Plex enables you to stream media such as films and music to other devices

Project Idea: Web Server

With built-in Wi-Fi (and an Ethernet port on some models) and a Linux operating system, Raspberry Pi is perfect for running your own web server. Build your own LAMP (Linux, Apache, MySQL, PHP) server with this Raspberry Pi Foundation tutorial: **magpi.cc/piwebserver**. The guide covers installing WordPress so you can host your own blog on the server.

A Raspberry Pi web server can also be used for self-hosting of a variety of services equivalent to third-party ones – particularly useful if you're worried about how your data is used by large corporations. Examples include PhotoPrism (photos), Nextcloud (documents), FressRSS (news), and Audiobookshelf (audiobooks).

Setting up your own Plex media streaming server is another option. See part three of our Ultimate Home Server guide in issue 110 (**magpi.cc/110**) for more details.

Project Idea: Weather Station

By connecting electronic sensors to Raspberry Pi's GPIO pins, you can use it to monitor the environment – see our Sensory World series starting in issue 111 (magpi.cc/111).

One popular educational project is to build your own weather station. While it's possible to create one using separate components and sensors, Pimoroni's Weather HAT (**magpi.cc/weatherhat**) makes the process a lot simpler. It features on-board sensors to monitor temperature, pressure, humidity, and light levels, whose readings can be displayed on the board's mini LCD. In addition, you can plug in optional external sensors to measure wind speed, wind direction, and rainfall.

You can even upload the live data to a web dashboard that you can view from anywhere. Check out our step-by-step tutorial in issue 119 (**magpi.cc/119**).

Learning Resources

A huge number of useful Raspberry Pi resources are available in book form or online, to help you learn at your own pace. Here are just a few to get your started.

Raspberry Pi Projects

The Raspberry Pi Foundation's learning resources website is an excellent place to get started with programming on Raspberry Pi and using it in a wide range of electronics projects. You can filter projects by topic, hardware, software, and difficulty level. Enjoy creating a laser tripwire, a burping jelly baby, and a pixel pet, just to name a few.

>> *magpi.cc/projects*

The Pi Hut Tutorials

As well as selling a massive array of Raspberry Pi-related products, The Pi Hut has an extensive learning resources section on its website, filterable by hardware or programming language. You'll discover a whole load of useful Raspberry Pi tips, tricks, and guides, along with how to connect and use various electronic components.

>> *magpi.cc/pihutguides*

Raspberry Pi Books

We don't like to boast, but we've produced numerous books to help you on your Raspberry Pi learning journey. Titles include *The Official Raspberry Pi Beginner's Guide*, *Retro Gaming With Raspberry Pi*, and *The Official Raspberry Pi Handbook*. You can buy the physical books or download PDFs.

>> *magpi.cc/books*

You can upload your weather data to a web dashboard

CrowVision

▶ Elecrow ▶ **magpi.cc/crowvision** ▶ TBA

A flexible, large touchscreen for all Raspberry Pis, and even your PC. **Rob Zwetsloot** has a look at it

SPECS

SCREEN:
11.6″, 1366×768, 178 degree IPS, five-point touch

I/O:
1 × mini HDMI, 1 × USB 2.0 power port, 1 × USB micro port for touch output, 1 × 3.5 mm audio jack

COMPATIBLE RASPBERRY PI:
Raspberry Pi 1, 2, 3, 4, Zero, Zero 2

Traditionally, a lot of touchscreen displays or general display add-ons for Raspberry Pi tend to be quite small as a sort of reflection of its size. While that's great and we definitely have a few projects that make use of these small screens, sometimes we need something a little bigger. Like this 11.6″ CrowVision, which lets you easily mount a Raspberry Pi to the rear of the display and just… plug it in.

This CrowVision has a bright and nice-looking IPS display, with decent viewing angles and plenty of easy mount options for putting it on a wall, or even just letting it free-stand on a surface, with some simple stands you can print off. While it has a bare PCB like a Raspberry Pi, you can find STL files to create a shell for it to make it look a bit nicer, wherever you intend to put it. Of course, if you're going that far, you can easily make one for yourself as well.

Slim and flexible

There is a small price to pay for the nice screen with touch capabilities, and that is with resolution. Perhaps we've been spoiled by modern consumer electronics, but 1366 × 768 does not seem like a lot of pixels, especially on a screen this size in 2023. It's definitely very functional and it's not like you're going to get a bad experience because of it, although that does largely depend on the kind of use case.

Kitchen computer or smart-home controller? Absolutely fine. As a screen to watch media or play video games? Not the very best solution.

Speaking of different use cases, the screen is also platform-agnostic. It displays anything that will connect to the HDMI port on the back, and works as a nice auxiliary monitor for a PC in certain situations. We like the suggestion of a little screen on your case full of Windows Media

▼ The screen supports just about anything with a video out port

There are plenty of good example use cases

Player visualisations, although something to keep track of a chat, or social media, is also handy. You can even use it this way with a Raspberry Pi – Raspberry Pi 4 and 400 do have dual monitor out after all.

See it all

In actual use, the touch is nice and responsive, and that aspect just requires you to plug in a USB cable to Raspberry Pi one end, and into the micro USB port on the display. Setup is very straightforward – it is just a screen after all, and the touch uses standard drivers – and if you're not great at changing audio out settings on Raspberry Pi, a handy 3.5mm jack on the screen has you covered for hearing what's going on. It even has a nice Realtek audio chip for the sound.

> In actual use, the touch is nice and responsive, and that aspect just requires you to plug in a USB cable

It's good, it's very practical, and it's one of the rarer bigger screens for Raspberry Pi. We recommend.

Simply mount a Raspberry Pi to the rear of the display with the included standoffs and screws

Verdict

While not the highest pixel density, it's a good-quality screen that is easy to set up and fills a bit of a niche.

8/10

Pironman

SunFounder ▶ **magpi.cc/pironman** ▶ £55 / $70

SPECS

CASE:
Aluminium tower case for Raspberry Pi 4 Model B (not compatible with Raspberry Pi 3)

INTERNALS:
Ice tower cooler with 5 mm copper pipe; Silent RGB fan with temperature control; On-board USB to M.2 SATA SSD, Supports TRIM and UASP; Available SSD Lengths: 22 mm × 30 mm, 42 mm, 60 mm, and 80 mm; Programmable 0.96″ OLED

I/O:
IR receiver for media centre; Power button for safe shutdown

Verdict

A complex build results in a charming, if slightly hacky, micro PC-tower-style case with a range of interesting components.

8/10

This detailed tower case is packed with features and a complex build. By **Lucy Hattersley**

Pironman features an OLED display on the front and RGB lighting inside

Pironman is a micro tower-PC case for Raspberry Pi 4, which it surrounds it with aluminium and acrylic metal plates. The assembled product is reminiscent of a PC shrunk down to fit your hand.

It is packed with interesting components, including an 0.96-inch OLED display, an ice cooler tower with 5mm copper pipe, a fan, an RGB light strip, an on-board USB to M.2 SATA SSD port, a power button, and an IR receiver. This is backed up by a software script that manages the OLED screen, internal lighting and fan.

Putting it together

The build process for Pironman is long and fiddly. The case itself comes partially assembled, but immediately requires disassembly to fit your Raspberry Pi inside. There then follows a 24-step assembly process that involves multiple flat conductor cables (FCC) to wire together the OLED display, SD card extender, and GPIO connector.

There are also multiple standoffs to connect the Pironman board underneath your Raspberry Pi 4 board. In total, we counted 28 different types of component in the case.

> "Packed with components with a unique charm"

There are multiple opportunities for the build to go wrong. In this instance, however, ours booted up and worked the first time. Still, the build took us around three careful hours to complete.

The end result is cute, in a slightly hacked-together way with plenty of screws and gaps. It's nowhere near as polished as some of the other cases on the market. But it is packed with components with a unique charm.

The finished tower is a tiny PC-tower-style case

SFERALABS

Raspberry Pi goes industrial

Long term availability, high reliability and support

Tailor-made embedded solutions

Our range of Raspberry Pi-based devices is wide. Really wide.
But what if your project needs something more? We can put all of our expertise into your custom OEM design.

Visit our website www.sferalabs.cc

POWERED BY Raspberry Pi

ONLY THE BEST

Spring is here! Time to deploy the sensors!

A collection of electronics for outdoor remote sensing

By **Marc de Vinck** @devinck

I've covered gardening-related electronics a few times, including wireless sensors, and a host of outdoor-related devices. And since spring is here, it made me think it's time to look at the possibilities of remote sensing again. I have always wanted to bring a little IoT to my own garden, and hopefully this is the year I do implement some kind of electronics. The past few years have been difficult to tackle that project since components, especially microcontrollers, have been difficult to come by, but that's finally changing. You can source microcontrollers and some single-board computers of various flavours at most of the online shops, and even a few brick-and-mortar retail shops. And I'm seeing new sensors being made available every week that would be perfect for remote monitoring.

Even as I write this article, I can hear the birds chirping outside, and I have a view of a very sad-looking garden that is just starting to show signs of life after its winter hibernation. I'd like to start gathering data, like sun exposure, moisture levels, and possibly even a camera to catch any critters raiding the tomatoes. So that's where this Best of Breed comes into play. I'll be looking at a few things that I'd like to add to my garden this summer to help me go from a boring plot of land to one that can start to speak to me about its needs.

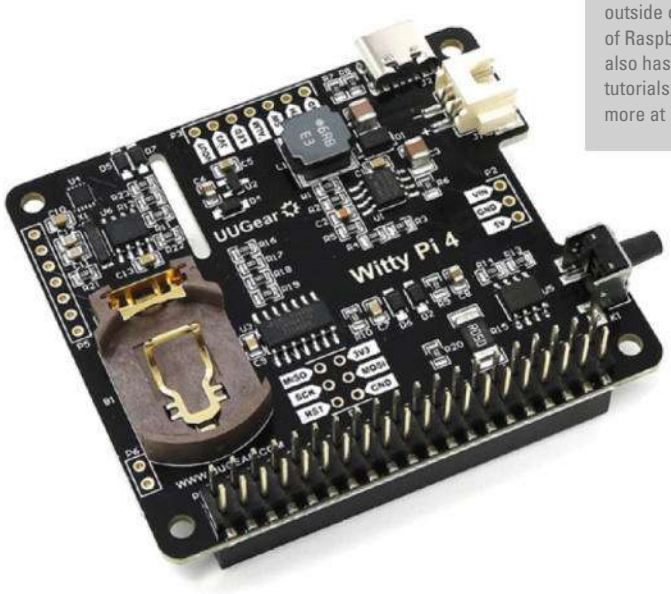
Witty Pi 4 vs Adafruit Feather RP2040 with RFM95 LoRa Radio

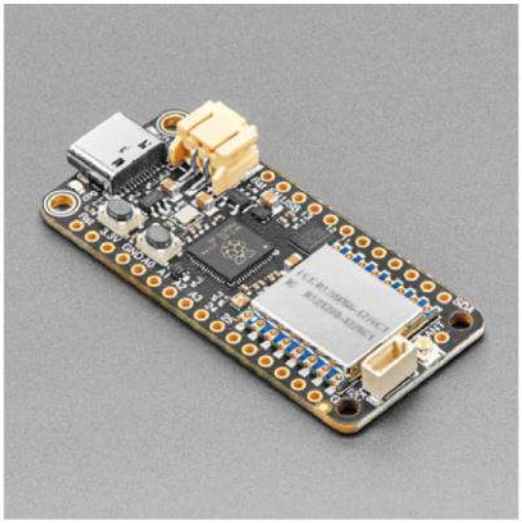
UUGEAR $27.63 | hsmag.cc/WittyPi4

ADAFRUIT $29.95 | adafruit.com/product/5714

The Witty Pi 4 from UUGear is a great solution for all your Raspberry Pi timing, scheduling, and power management needs. No longer will you need to worry about the accuracy of your Raspberry Pi clock thanks to the on-board real-time clock and temperature sensor that allows for additional accuracy and temperature compensation. And anyone who has worked with the Raspberry Pi knows about power management and the relatively difficult nature of powering your Raspberry Pi on or off, and this is where the Witty Pi really shines.

Unlike a microcontroller that you simply power-off, you need to go through a powering-off sequence with your Raspberry Pi, similar to your desktop computer. And the Witty Pi can do that too! Now your Raspberry Pi can shut down and only wake up when needed, saving lots of power. If you have a remote monitoring project, or one that runs off battery power, the Witty Pi should be something to consider adding to the build.





The Adafruit Feather RP2040 RFM95 LoRa Radio board, aka 'RadioFruits', is Adafruit's take on a microcontroller with integrated long-range (LoRa) packet radio transceiver, along with a built-in USB and battery charging system. This is the perfect microcontroller board for creating wireless sensors that have a longer range and require less power than other wireless solutions like Bluetooth.

The board features an RP2040 chip running at 133MHz, the same as you can find on a Raspberry Pi Pico. You also get a generous 8MB of on-board QSPI flash and 264kB of RAM, plus on-board RGB LED, STEMMA QT connector, and lots of GPIO pins. Between the on-board battery management, LoRa radio, and RP2040, it would be hard to beat this little board for creating an outdoor sensor network.

VERDICT

Witty Pi 4

Great addition to a Raspberry Pi.

9/10

Adafruit Feather RP2040 with RFM95 LoRa Radio

Simple long-range sensing.

10/10

HackSpace

This articlel is from HackSpace magazine. Each issue includes a huge variety of maker projects inside and outside of the sphere of Raspberry Pi, and also has amazing tutorials. Find out more at **hsmag.cc**.

M5Stack PoE Camera with Wi-Fi (OV2640)

PIMORONI £39.90 | hsmag.cc/M5Stack

The PoECAM-W is an open-source and programmable camera that integrates PoE (power over Ethernet) and Wi-Fi giving you plenty of connectivity options. The brain of the product is an ESP32 board with an integrated webcam and W5500-embedded Ethernet controller. You get a 2-megapixel image sensor, 8MB PSRAM, and 16MB flash large memory.

This could be a good addition for monitoring your outdoor space for pesky intruders, capturing time-lapse videos of the growing season, or just seeing what's going on outside. It's easy to deploy since you can program it via the Arduino IDE or ESP-IDF. And, of course, you must consider how you will make it waterproof, but that could be as simple as mounting it to the inside of a window in your house or shed.

VERDICT

M5Stack PoE Camera with Wi-Fi (OV2640)

The perfect compact camera.

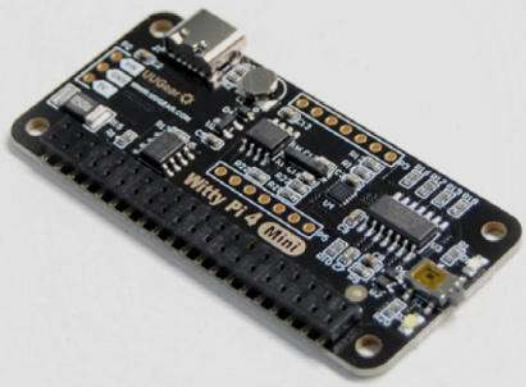
10/10

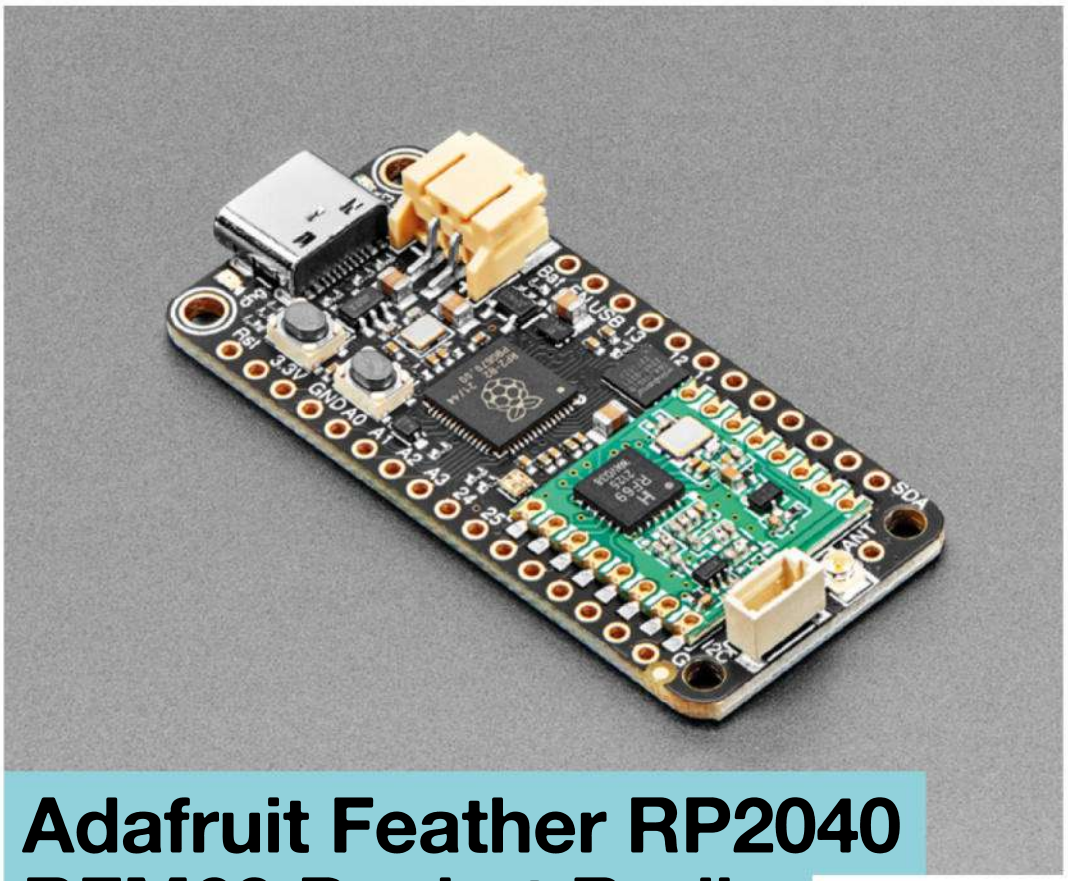
WITTY PI 4 MINI

UUGEAR $18.42

hsmag.cc/WittyPi4Mini

Just like its big sibling, the Witty Pi 4 Mini is a great addition to your Raspberry Pi, but in a smaller package and with fewer options. The Witty Pi adds a real-time clock with power management to your Raspberry Pi. It can control your Raspberry Pi's on/off sequence, greatly reducing its power usage. If you've worked with a Raspberry Pi before, you know the challenges of power management, and this board does a great job!



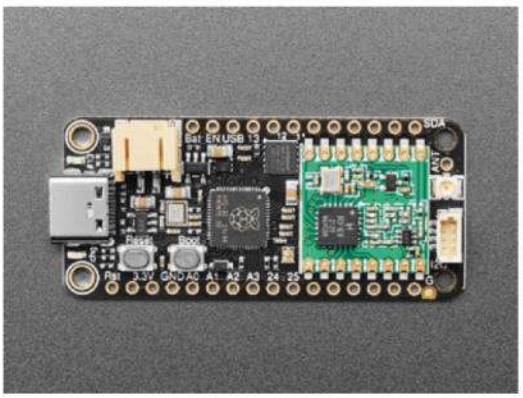


Adafruit Feather RP2040 RFM69 Packet Radio

ADAFRUIT $19.95 | adafruit.com/product/5712

Left
Remotely control your garden

The Adafruit Feather RP2040 RFM69 board is another board Adafruit lovingly refer to as part of its 'RadioFruits' line of microcontrollers. The board is very similar to the RP2040 with LoRa, but this one features an RFM69 Packet Radio (868 or 915MHz). It's great for sending packets of data between different boards featuring the same type of radio. You can communicate up to five kilometres, but of course that's under ideal conditions and with a proper antenna. Speaking of antennas, you will need one for this board to work, and you can make your own with a small piece of wire, or you can buy one and use the on-board connector.



VERDICT

Adafruit Feather RP2040 RFM69 Packet Radio

A good low-cost radio and micro.

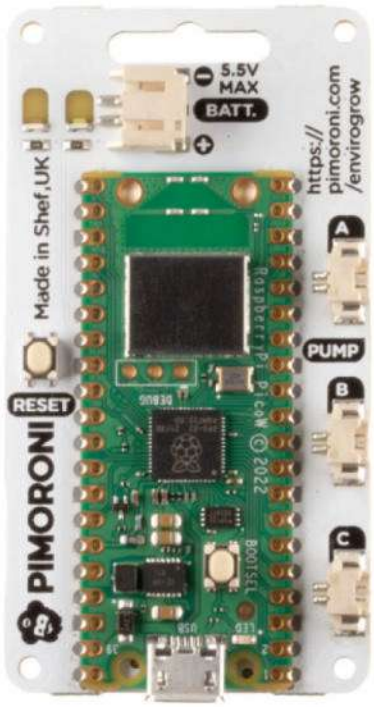
9/10

Enviro Grow (Pico W Aboard)

PIMORONI | $51 | hsmag.cc/EnviroGrow

And how can we not cover the Enviro Grow systems by Pimoroni when talking about remote sensing and gardening? In a previous Best of Breed, I reviewed the Grow Kit, and no surprise, it was great. Well, now there is the Enviro Grow kit with a Raspberry Pi Pico on-board, featuring lots of accessories to get you started.

The kit includes the main Enviro Grow board with the Raspberry Pi Pico on the back. The board also includes an on-board BME280 temperature, pressure, humidity sensor, a Piezo buzzer, and a light and proximity sensor. And that's just the beginning of what's included in the kit – you also get three moisture sensors, three mini submersible pumps, a battery pack, batteries, and a bunch of other parts to get a fully automatic watering system up and running. It's a great kit for a great price. Head over to the Pimoroni site to learn more.



Left
Get your plants to water themselves

VERDICT

Enviro Grow (Pico W Aboard)

Environmental sensing with a Pico!

10/10

Expand your Raspberry Pi

Stackable expansion boards for Raspberry Pi

RS485 Pi

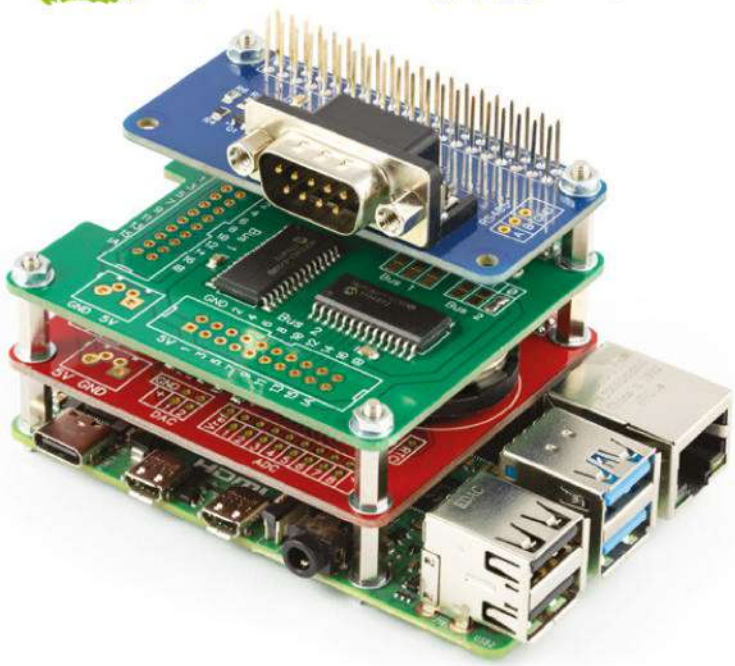
RS485 serial communication board.

IO Pi Plus

32 channel digital I/O controller.

Expander Pi

16 digital I/O channels, 8 analogue inputs, 2 analogue outputs and a real-time clock.



ABelectronics UK

www.abelectronics.co.uk

TILT
A wireless hydrometer and thermometer that connects to your Raspberry Pi
Free international shipping with checkout code:
SHIP-FREE-INTL
shop now at tilthydrometer.com
Download SD Card Image here

10 Amazing:

Camera projects

Are you getting the shot, or is the shot getting you?

Raspberry Pi Camera Modules are incredibly versatile. Whether you're making your own custom camera or creating an automated motion sensor camera, there's so much you can do thanks to the inbuilt libraries. And it's not just Camera Modules you can do it with. Here are just some of our faves...

Pi Cam

All-in-one camera

This handy-dandy camera can be taken around with you to snap photos, used to monitor somewhere with remote video, or take time-lapses! Very smart.

magpi.cc/picam

3D camera

In stereoscopic

While the early 2010s' 3D fad has sorta died down, the current virtual reality trend has not. With this cool tutorial, you can take your own 3D photos or video – with a little practice.

magpi.cc/3dcamera

Raspberry Pi 3D Scanner

Model yourself

Apparently one of Hollywood's most popular 3D actor scanners is a truck filled with Raspberry Pi. It uses tech a lot like this. Perhaps you can become the new most popular actor scanner.

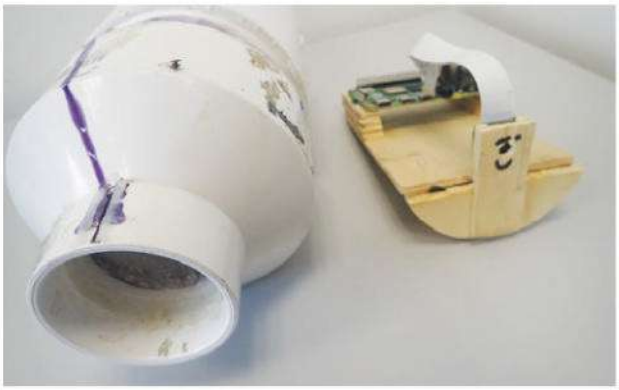
magpi.cc/3dscanner

Bullet time video booth

Free your mind

Relive the bit in *The Matrix* where Neo wiggles to the ground, but in your very best tux/dress, and no wires to help you.

magpi.cc/bulletwedding

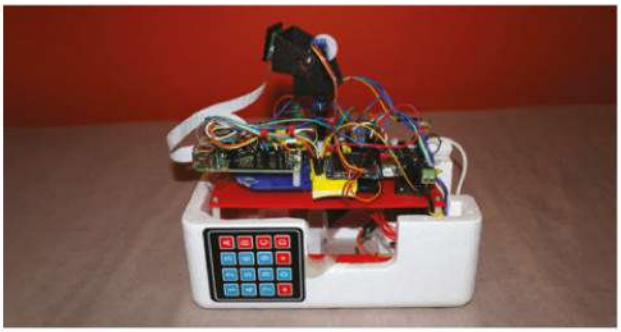


Antarctic Picam

Under the ice

This ingenious little device uses a simple plumbing seal to make a water-tight device that has been used to study algae on polar ice.

magpi.cc/130



Motion time-lapse camera rig

Move slowly

We haven't talked about this tutorial in a while, which helps you create carefully controlled moving time-lapses that can sweep an area.

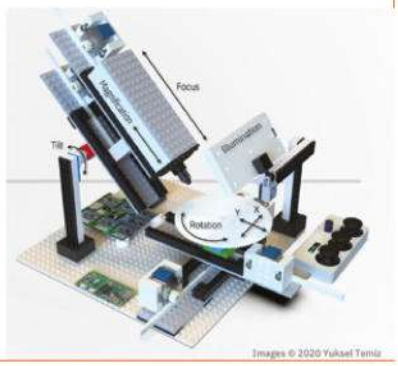
magpi.cc/pantiltlapse

Homemade microscope

LEGO miniatures

This device is stunning, being able to easily photograph the edges of a 1p coin with incredible clarity, and it's built out of LEGO to boot.

magpi.cc/diymicro



PiFinder

Watch the skies

This doesn't help you find Raspberry Pi computers (that's rpilocator), this helps you find – and track – celestial bodies for better astrophotography and stargazing.

pifinder.io

Pieca

Old school lens

Using classic Leica camera lenses with Raspberry Pi High Quality Camera creates a beautiful anachronistic device that takes lovely photos.

magpi.cc/pieca

Naturebytes

Wildlife photography

An incredibly easy way to set up your own camera trap for nature in your garden. Pop in a NoIR Camera Module and some IR LEDs, and you can get some great shots of night critters too.

naturebytes.org

Learn Linux with Raspberry Pi

Resources to help you delve into the Linux operating system.

By **Phil King**

The Linux Command Line

AUTHOR

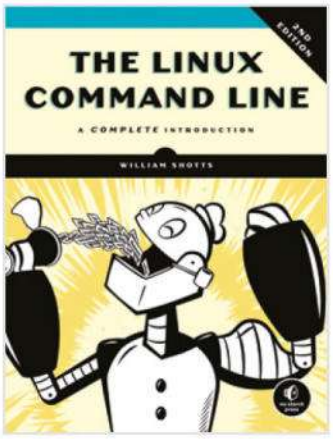
William Shotts

Price:
$39.95 / Free (PDF)
magpi.cc/linuxclbook

While most Linux distros, including Raspberry Pi OS, offer the option of a GUI, the command line is where you can really get into the internal workings of the system and use powerful text-based commands to control every aspect.

One of the best resources for learning about it is William Shotts's comprehensive 555-page guide, which covers the same material as the **LinuxCommand.org** website, but in much greater detail.

The free PDF, now in its fifth edition, is divided into four main sections. Part one covers topics such as navigating the file system, manipulating files and directories, permissions, and processes. Part two introduces text editors, with a detailed section on vi, and prompt customisation. Part three covers common tasks and essential tools such as package management and regular expressions. Finally, part four is all about writing Shell scripts.



Bookmark these

Learn Linux using these websites

LINUX SURVIVAL

With an in-browser command-line interface window, this tutorial aims to make it as easy as possible to learn Linux. It covers the most useful commands and includes a quiz at the end of each module.

▶ linuxsurvival.com

COMMAND LINE FU

This searchable repository of Linux terminal commands is a goldmine of information. Users share their most useful commands, as can you. It can also be used to store your own favourites.

▶ commandlinefu.com

DEBIAN WIKI

Since Raspberry Pi OS is built on top of Debian Linux, this wiki should come in very handy. It comprises a number of 'portals' on aspects such as system administration and networking.

▶ wiki.debian.org

Linux Journey

AUTHOR
Cindy Quach

Price:
Free

linuxjourney.com

One of the most popular operating systems in the world, Linux comes in a wide range of varieties, known as 'distributions' or 'distros'. Raspberry Pi OS is based on the Debian Linux distro. Therefore, learning more about Linux can help you get under the hood of your Raspberry Pi, understand how it works, and make the most of its capabilities.

Available in several languages, the Linux Journey website is a great place to start. Billed as helping you to 'learn the ways of Linux-fu', it's organised into levels of expertise – from Grasshopper, though Journeyman, to Networking Nomad. Each level has a number of lesson modules to explore different aspects of Linux.

Topics range from the command-line interface, text manipulation, running processes, installing packages, and navigating the file system, to networking aspects such as DNS and routing. Each module comprises a series of steps, some with exercises (to run on your real Linux system) and a quiz.

Introduction to Linux

AUTHOR
edX

Price:
Free

magpi.cc/edxlinux

Created by the Linux Foundation, this free online course (with optional paid upgrade for graded assignments and a certificate) is an easy-to-follow introduction to the operating system. While you're free to learn at your pace, you can set a weekly goal for extra motivation.

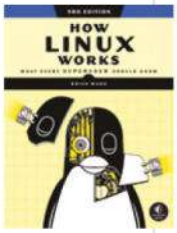
The course focuses on the three major Linux distro families: Red Hat, SUSE, and Debian. It doesn't cover the Debian-based Raspberry Pi OS specifically, which can cause a little confusion with details such as GUI settings.

After a couple of chapters on the Linux Foundation and the Linux philosophy, you move onto learning you way around a typical Linux system. While it doesn't go into great detail on each topic, the course does give a good overview for beginners.

Reading list

Books to help you learn Linux

HOW LINUX WORKS



Subtitled 'What Every Superuser Should Know', Brian Ward's 464-page tome is very thorough, delving into Linux's inner workings, yet well-structured for dipping into whenever you need it.

magpi.cc/howlinuxworks

LINUX BIBLE

Now in its tenth edition, this comprehensive Linux user guide requires no prior knowledge, but is packed with essential info. With an emphasis on command-line tools, it's largely distro-agnostic.

magpi.cc/linuxbible

THE DEBIAN ADMINISTRATOR'S HANDBOOK

Written by two Debian Linux developers, this is a superb resource for anyone using a Debian-based distro such as Raspberry Pi OS or Ubuntu. Read it online, download the PDF, or buy the book.

debian-handbook.info

HannahMakes

A journalist turned special effects technician, you'll find **Hannah** making great stuff on YouTube

- Name HannahMakes | Occupation Special Effects Technician
- Community role YouTuber | URL hannahmakes.com

Sometimes you find out that you've been training your whole life for something, and it becomes a fantastic new hobby. For us, and many of you reading the mag, that ended up being making.

"I grew up really fond of crafts and obsessed with tech, but I didn't really figure out how to put the two together until much later," Hannah explains. She makes excellent videos, as HannahMakes, about her wonderful contraptions. "I happened to be working for New Scientist magazine when I started hearing things about the maker movement; people using a blend of traditional techniques and tech to build awesome things at home. It felt like something we should be covering at the magazine, so I kept mentioning how I thought someone should write about it. My colleagues must have got fed up of hearing me whine, because eventually they told me that, if it bothered me that much, perhaps I should write about it. So I did!"

Hannah ended up buying a 3D printer and an electronics kit and jumped in the deep end, staying up late to figure out how to get them all working.

"I wrote a column about the crazy projects I came up with, like teabag-dunking robots, and shoes that call you an Uber when you click your heels together three times," Hannah continues. "As time went on, I became so obsessed with making that I ended up switching careers to find a job that involved less writing and more building. Now I get to work in a giant workshop full of machines (my favourite is probably the giant 7-axis robot arm), and wonderful, talented people that I get to learn from."

Hannah found a passion for making videos at New Scientist

Where did your YouTube channel come from?

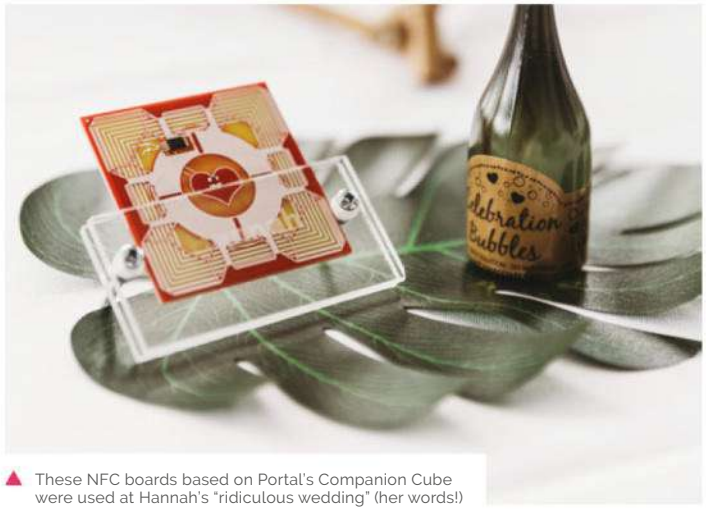
I'd made a few videos for New Scientist magazine and discovered I really enjoyed it – so much so that I wanted to continue doing it after I left the magazine. So, I started my own YouTube channel. I also liked the idea that showing off my projects might help other women to consider tech as a viable hobby or career. Despite an increased focus on inclusivity and diversity, the sad truth is that women are still a minority in the industry (Tech Nation recently reported that "just 26% of the tech workforce are women.") Besides, it can be an intimidating field with a relatively high barrier to entry, regardless of your gender.

So, I hope that by showcasing my work I can help show other women there's a place for them.

What are your favourite things you've made with Raspberry Pi? One of my favourites was a prop for a TV show. The production team gave us a regular digital alarm clock and asked us to make an exact replica, except an inch bigger and fully controllable. Before working in the film industry, I might have assumed you'd just use a normal alarm clock, but they wanted close-ups of the clock face showing a very specific time and advancing by one minute precisely on cue. There were also LEDs in the translucent body, and they wanted to control the colour and brightness. In the end, we 3D-printed a case and I fitted in a Raspberry Pi running a small LED strip and a Pimoroni HyperPixel touchscreen. I wrote a program using Pygame to display a seven-segment-style clock, and added an invisible button to one corner of the screen, which led to a menu where you could set the time and other options. I also added some buttons on a long cable so they could step forward and backward through time as they needed throughout a scene. It was an extra-memorable project because, once we were finished, my boss let me blow up the original alarm clock in the car park with some pyrotechnics we were testing!

> Now I get to work in a giant workshop full of machines and wonderful, talented people

The other project that comes to mind is kind of a bit naughty: I used a Raspberry Pi to hack an old 3D printer. The printer would only work with proprietary cartridges of branded filament that were expensive and tricky to get hold of. The cartridges contained EEPROM memory to store info like serial number and remaining material, which the machine would read, preventing you from opening a cartridge and filling it with your own material. But, some kind geniuses cracked the encryption used on the EEPROM chips, meaning that all you need is a Raspberry Pi and a couple of croc clips to reprogram a cartridge and trick the printer into thinking it's brand-new (**magpi.cc/staratools**). I think it's justified because it gave a new lease of life to a machine that wasn't seeing much use any more, and let us experiment with some new materials.



▲ These NFC boards based on Portal's Companion Cube were used at Hannah's "ridiculous wedding" (her words!)

▲ The Commonwealth Games bull was programmed by Hannah

▼ Bull head, with Hannah for scale

MagPi Monday

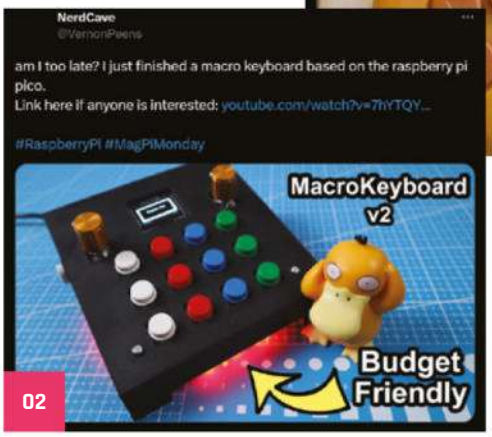
Amazing projects direct from social media!

Every Monday we ask the question: have you made something with a Raspberry Pi over the weekend? Every Monday, our followers send us amazing photos and videos of the things they have made.

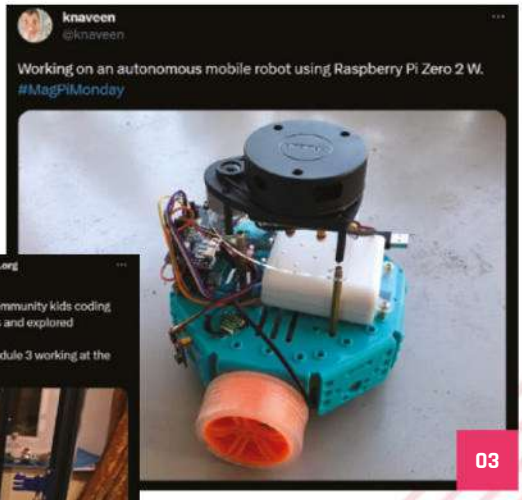
Here's a selection of some of the awesome things we got sent this month – and remember to follow along at the hashtag #MagPiMonday!

Kevin McAleer Robot Maker
Happy #MagPiTuesday! I recently made a Chicken Nugget run Doom, using a Raspberry Pi Zero 2W, and a @pimoroni Display HAT Mini.
Writeup here: kevsrobots.com/blog/chicken_n...
@McDonaldsUK #raspberrypi #doom

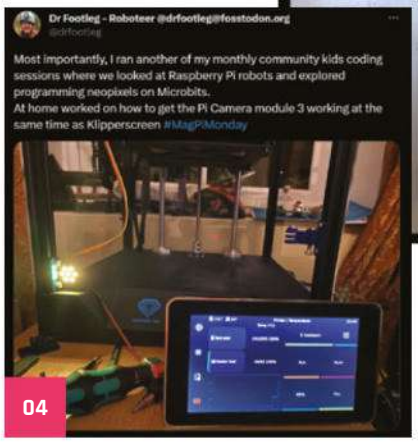
01


NerdCave
@VernonPeens
am I too late? I just finished a macro keyboard based on the raspberry pi pico.
Link here if anyone is interested: youtube.com/watch?v=7hYTQY...
#RaspberryPi #MagPiMonday
MacroKeyboard v2
Budget Friendly

02



03


Dr Footleg - Roboteer @drfootleg@fosstodon.org
Most importantly, I ran another of my monthly community kids coding sessions where we looked at Raspberry Pi robots and explored programming neopixels on Microbits.
At home worked on how to get the Pi Camera module 3 working at the same time as Klipperscreen #MagPiMonday

04

01. DOOM on a Chicken McNugget is a sentence we'd never even considered
02. It's astonishingly easy to make a macro keyboard with Raspberry Pi Pico
03. This robot looks like it's packing everything but the kitchen sink
04. Raspberry Pi robots and a 3D print controller? An excellent weekend for Dr Footleg
05. This is a heavy-duty Raspberry Pi project list! Practical projects and fun projects combined
06. Think we've seen that pirate ship on Power Rangers
07. Mike Horne's cyberdeck project is filling out!
08. The weather station will be very confused in this British 'summer'
09. We love this and desire one for international business trips
10. Unfortunately, this digger cannot read *The MagPi* to you
11. Meteor showers are very cool when you can see them. An excellent build

05

Yukari-Sensei
@YukariHx0

Project 1: Mini NAS 6TB
Project 2: Docker (Plex, piHole, etc)
Project 3: Arcade console

rPi's, switch, storage, adapters, etc is under my desk.

Got to arrange wires and replace a power source to have a cleaner setup, but ~70% is done so far

06

Leo White
@LeoWhitesTweets

I used a @StereoPi combined with python code from @orionrobots book to stream live video from my robot to an Oculus quest headset so I didn't have to run around the @Hacky_Racers course this weekend!

Mike Horne
@recantha

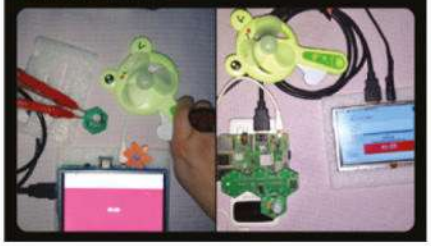
Hi #MagPiMonday, this weekend I decided to drill the lid of my mini-cyberdeck in order to secure the screen. I'm just waiting on a Bluetooth mini keyboard now to reduce the amount of cabling in the box! That's the #RaspberryPi in the bottom right, to give you a scale!

07

08

Aula Jazmati
@Aula_J2018

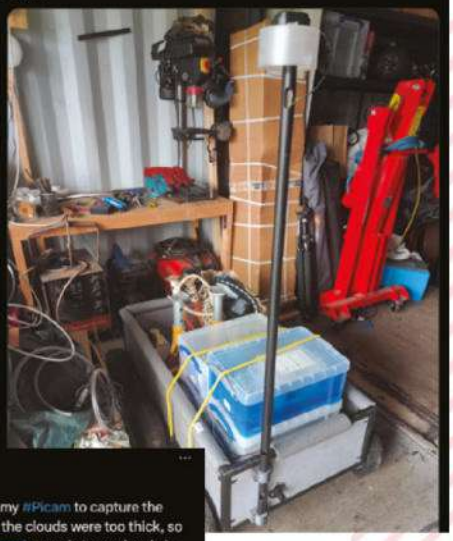
G🟡🟡d Morning 🌿
My new project: Simple #raspberrypi weather station with @Hexabitzinc ✨
hackster.io/aula-jazmati/m...



09

Keegan Neave
@NeaveEng

More work on The Luggage, a cargobot that will be able to follow me around



10

Roland Schulz
@r_schulz_maker

Work in process. I try to grab a hardcopy of the latest @TheMagPi magazine with my DiggerRobot. Turn a kids toy into a robot with encoder motors, huge servos and @pimoroni inventor hat mini on a @Raspberry_Pi Zero W. All tests passend when run examples👍💪☎️ #MagPiMonday

Pierre-yves Baloche
@FunkyFlwy

Happy #MagPiMonday to all! I tried to use my #Picam to capture the #Perseids this weekend but unfortunately the clouds were too thick, so ended up building a #flightaware #raspberrypi console to see local air traffic. There's always a #backupPiProject to do 😁

11

Events in pictures: Raspberry Pi Community Gathering Taiwan

Community and official events in the wild

Held regularly in Taiwan, these community gatherings have people come and show off their projects and talk about how they work. This recent event, from the end of July, also featured friend of the magazine Masafumi Ohta from the Japanese Raspberry Pi User Group.

FIND OUT ABOUT NEXT MONTH'S EVENTS AND POP-UP ON PAGE 94

01

02

01. Car computers powered by Raspberry Pi were on show
02. A wide range of projects on display – including the weather station kit
03. Masafumi guests at the Community Gathering
04. Free, official Raspberry Pi swag is available for folks that put on Raspberry Jams and related events

04

03

Best of the rest!

Other cool things we saw using Raspberry Pi this month

Memory Box

This is apparently the creator's first project, and it looks great. It's a wooden framed 'Memory Box' which responds to NFC cards to show videos of specific memories. It's easy to disassemble for maintenance, and makes a nice little bedside clock too.

▶ **magpi.cc/membox**

BMO upgrades

We featured the BMOctoprint a while back in the magazine. It's very cool and cute. Allie has recently upgraded it with a Jake the Dog (another character from *Adventure Time*) themed carrier seat so they can take it DragonCon.

▶ **katzcreates.com**

Together we can make a **difference**

Give young people the opportunity to learn about technology

The Raspberry Pi Foundation enables young people to realise their full potential through the power of digital technologies, but we can't do this work without your help. Your support helps us give young people the opportunities they need in today's world. Together we can offer thousands more young people across the globe the chance to learn to create with digital technologies.

Generous donations from organisations and individuals who share our mission make our work possible.

DONATE NOW

Donate today to make a difference: **rpf.io/magpidonate**

Raspberry Pi Foundation

Your Letters

#MagPiMonday: how to be in the magazine

I've been wanting to share something for #MagPiMonday, and I was wondering if there are any specific requirements to get featured in the magazine section with pictures of them?

Jude via Threads

We don't have any specific requirements to get into This Month in Raspberry Pi – as long as you have a nice photo of your project and we think it's cool, it will usually make its way in. The limiting factor mostly is page space. We try to limit it to one post per person in an issue too.

Either way, it's mostly a way for you to show it off on social media.

#MagPiMonday: other social media

I'm moving away from Twitter or X, or whatever it will be called next week. Will I still be able to send in #MagPiMonday submissions? Are you on any other social media platforms?

Amar via email

We've revived our Mastodon account (**magpi.cc/mastodon**) for just this reason – and we're now getting a little more active on Threads and Instagram (we're **@themagpimag** on both of those). If you really want to show us something cool, you can always email us about it too!

Large slice of Raspberry Pi

I saw the giant Raspberry Pi near the Raspberry Pi Store in the Cambridge Grand Arcade – does it actually work? Will I be able to buy one? Might make for a good kit...

Cam via Twitter

Unfortunately it's a one-off built by Raspberry Pi's Maker in Residence Toby Roberts (who is also a regular in the magazine), so you won't be able to get a kit. At least, there's no plans for one yet. Also, it doesn't really work – a Pico controls some of the lights on it but, otherwise, it's just for show. You probably could make it work if you were really dedicated to it though!

▲ The giant LED also isn't quite a standard part

Contact us!

- Twitter **@TheMagPi**
- Facebook **magpi.cc/facebook**
- Email **magpi@raspberrypi.com**
- Online **forums.raspberrypi.com**

USA SPECIAL!
6 ISSUES
FOR $43

YOUR OFFICIAL RASPBERRY PI MAGAZINE
The MagPi
The official Raspberry Pi magazine
UNIVERSAL MEDIA PLAYER
Save money by building your own streaming box
GET INVOLVED WITH CITIZEN SCIENCE
BUILD A BAT DETECTOR
TOP 10 UPCYCLING PROJECTS
PICO RETRO GAMING WITH KITRONIK ZIP96

FREE
RASPBERRY PI
PICO W

Subscribe online:
magpi.cc/subscribe

Continuous credit card orders will auto-renew at the same price unless cancelled. A free Pico W is included with all subscriptions. This is a limited offer. Not included with renewals. Offer subject to change or withdrawal at any time.

Community Events Calendar

Find out what community-organised Raspberry Pi-themed events are happening near you...

01. Melbourne Raspberry Pi Meetup

Sunday 3 September
Docklands Makerspace and Library, Melbourne, AUS
magpi.cc/mrpm133

This meetup is open to everyone with an interest in electronics, robotics, home automation, 3D printing, laser cutting, amateur radio, high altitude balloons, space tech, etc. Makers are invited to bring along their projects and project ideas, and come connect with other makers. Get your questions answered, show off the work you are doing, and get support to resolve nagging issues.

02. Riverside Raspberry

Monday 11 September
3600 Lime Street, Riverside, CA, USA
magpi.cc/riverside133

Riverside Raspberry is a California-based enthusiast group that's been meeting up for almost a decade. This group will appeal to thinkers and tinkerers, the scientifically curious, software developers looking to get into embedded development, or anyone young or old that has a knack for tech in general.

03. Autumn Computer Fair 2023 (and AGM)

Saturday 23 September
University of Hertfordshire, Hatfield, UK
magpi.cc/acf23

Something old and something new? Not sure about borrowed or blue, but do read on... Join us this September for a morning of computing fun. We will have displays and hands-on opportunities with older working systems from the 1970s, 1980s, and 1990s, as well as modern kit builds, and hack projects such as with the Raspberry Pi, UH Robotics and even drones!

04. Raspberry Pint

Tuesday 26 September
Online
magpi.cc/pint133

Raspberry Pint is a meetup for sharing digital making experiences. Most presentations are about building personal or professional projects with Raspberry Pi and other maker technology. You can also learn skills and techniques such as website design, PCB design, software development, 3D printing, and soldering. All makers and all digital making welcome. The quirkier the better.

FULL CALENDAR

Get a full list of upcoming community events here:

magpi.cc/events

RASPBERRY PI AT MAKER FAIRE

Where can you find Raspberry Pi next?

- Next location Maker Faire Rome
- Where Fiera Roma, Rome, Italy
- When Friday 20 October to Sunday 22 October

Raspberry Pi is proud to partner with Approved Reseller Melopero at Maker Faire Rome again in 2023. Come and meet members of the Raspberry Pi team, learn about their latest products, and share what you've made with Raspberry Pi technology.

magpi.cc/mfrome23

WIN ONE OF FIVE RASPBERRY PI 4 8GB

To celebrate Raspberry Pi 4 being more widely available, we're giving away five of the 8GB version to some lucky readers. All we ask is that you make cool stuff with it.

Head here to enter: magpi.cc/win | Learn more: magpi.cc/raspberrypi4

Terms & Conditions

Competition opens on **30 August 2023** and closes on **28 September 2023**. Prize is offered to participants worldwide aged 13 or over, except employees of Raspberry Pi Ltd, the prize supplier, their families, or friends. Winners will be notified by email no more than 30 days after the competition closes. By entering the competition, the winner consents to any publicity generated from the competition, in print and online. Participants agree to receive occasional newsletters from The MagPi magazine. We don't like spam: participants' details will remain strictly confidential and won't be shared with third parties. Prizes are non-negotiable and no cash alternative will be offered. Winners will be contacted by email to arrange delivery. Any winners who have not responded 60 days after the initial email is sent will have their prize revoked. This promotion is in no way sponsored, endorsed or administered by, or associated with, Instagram, Facebook, Twitter or any other companies used to promote the service.

Halloween!

Frighteningly good builds for the spooky season!

THE MAGPI **#134**
ON SALE **28 SEPTEMBER**

Plus!

A giant Raspberry Pi truck printer

Build a Scoopy Oscilloscope

Control a model railway with Raspberry Pi

DON'T MISS OUT!
magpi.cc/subscribe

TWITTER @TheMagPi
FACEBOOK fb.com/MagPiMagazine
EMAIL magpi@raspberrypi.com

EDITORIAL

Editor
Lucy Hattersley
lucy@raspberrypi.com

Features Editor
Rob Zwetsloot
rob@raspberrypi.com

Sub Editor
Nicola King

ADVERTISING

Charlotte Milligan
charlotte.milligan@raspberrypi.com
+44 (0)7725 368887

DESIGN

criticalmedia.co.uk

Head of Design
Lee Allen

Designer
Sam Ribbits

Illustrator
Sam Alder

CONTRIBUTORS

David Crookes, Rosie Hattersley, Nicola King, Phil King, KG Orphanides, David Plowman, Mark de Vinck, Stewart Watkiss

PUBLISHING

Publishing Director
Brian Jepson
brian.jepson@raspberrypi.com

Director of Communications
Liz Upton

CEO
Eben Upton

DISTRIBUTION

Seymour Distribution Ltd
2 East Poultry Ave,
London EC1A 9PT
+44 (0)207 429 4000

SUBSCRIPTIONS

Unit 6 The Enterprise Centre
Kelvin Lane, Manor Royal,
Crawley, West Sussex, RH10 9PE
+44 (0)1293 312193
magpi.cc/subscribe
magpi@subscriptionhelpline.co.uk

This magazine is printed on paper sourced from sustainable forests and the printer operates an environmental management system which has been assessed as conforming to ISO 14001.

The MagPi magazine is published by Raspberry Pi Ltd, Maurice Wilkes Building, St. John's Innovation Park, Cowley Road, Cambridge, CB4 0DS. The publisher, editor, and contributors accept no responsibility in respect of any omissions or errors relating to goods, products, or services referred to or advertised in the magazine. Except where otherwise noted, content in this magazine is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Unported (CC BY-NC-SA 3.0).
ISSN: 2051-9982.

Learning by making

How a Raspberry Pi for work and play is helping Lucy learn

By Lucy Hattersley

I'm currently in the process of building a Pikon Camera. This is a homemade digital camera with a 3D-printed case, LCD screen, battery and – of course – Raspberry Pi (**magpi.cc/pikon**).

So far, I've created the 3D-printed case in two options: one for Camera Module 3 and one for High Quality Camera, and I've assembled all the parts. It charges and powers up well.

To a mother, all children are beautiful, but I'll be the first to admit my lovely camera is a little wonky.

The software side of things is a bit of a struggle at the moment as well. The 3.5-inch WaveShare screen has a resolution of just 480 × 320. This is fine as a preview window, of course, but it's smaller than the default preview window used by libcamera, and the two are stubbornly refusing to work together.

Still, that just means I need to dig down into the documentation (**magpi.cc/cameradocs**), and read the manual (**magpi.cc/picam2man**). I tried using Bard and ChatGPT to create working code, but it was all out-of-date or simply incorrect. I'm enjoying hacking through it all.

In the meantime, I've got it up and running with a direct Ethernet connection. This is handy, as I helped take some photos of Babbage (the Raspberry Pi mascot) for Part 5 of our Camera Module tutorial (page 54).

On the whole, I'm pretty happy with how my little camera is coming along. Even if it looks a bit shonky and doesn't fully work yet. That's not always the point. Making my own camera is a fun project, but it's also an important learning experience. You never quite learn as much as when making. I've learned more about the Raspberry Pi Camera Module and integration with Python and the picamera2 library, Raspberry Pi power requirements, and the windowing system over the last few weeks than I ever would just reading the documentation. Making remains quite humbling.

> It helps that a lot of my work involves writing about Raspberry Pi, so doing that with Raspberry Pi feels nicely circular

Live and learn

Another feature that really interested me this month is Work & Learn with Raspberry Pi (page 64). Raspberry Pi is increasingly becoming my go-to work computer, and I bought a 60% mechanical keyboard to keep my desk tidy and type better with Raspberry Pi.

It's always interesting to me that Raspberry Pi was designed as a computer for making and hacking, but is increasingly useful for everyday computing tasks. It's a perfect little Linux machine for doing the regular task of writing, editing, and commissioning all the other great people who work on this magazine. Of course, it helps that a lot of my work involves writing about Raspberry Pi, so doing that with Raspberry Pi feels nicely circular.

Learning has always been a fairly central part of my life. It's always felt important to me to be developing myself. Sometimes it's just personal development; sometimes on a professional front. But, as long as I end the day having a little more nounce than before, I can go to sleep happy.

AUTHOR

Lucy Hattersley

Lucy is editor of *The MagPi* and is currently swapping out Ubuuntu on her ThinkPad for Raspberry Pi OS while gently cooing the MacBook not to worry too much. Maybe it should worry a bit, mind.

magpi.raspberrypi.com

HiPi.io

HIGHPI PRO

The new case from the HiPi.io team

- Rapid tool-free assembly and disassembly
- Large internal volume for HATs
- Compatible with Pi 2/3/4
- Multiple lid options
- Passive & Active Cooling options
- Secure microSD card cover
- VESA mount support
- Molding-configurable output ports customizable for volume orders
- Printed logo for your branding

Available at these great Pi stores:

PiShop.us PiShop.ca PiShop.mx ThePiHut Pi-Shop.ch Welectron.

Contact your favorite Pi store if it's not listed here

HiPi.io

PiKVM

Manage your servers or workstations remotely

A **cost-effective** solution for data-centers, IT departments or remote machines!

PiKVM HAT
for DIY and custom projects

Pre-Assembled version

- Real-time clock with rechargeable super capacitor
- OLED Display
- Bootable virtual CD-ROM & flash drive
- Serial console
- Open-source API & integration
- Open-source software

Available at the main Raspberry Pi resellers

core ELECTRONICS · PiShop.ca · RPishop.cz · Welectron. · elektorstore · PIMORONI

Pi-Shop.ch · ThePiHut · PiShop.us · CloudFree · PiShop.mx

Reseller suggestions and inquiries:
wholesale@hipi.io